

Accelerometer-based Biometric Gait Recognition for Authentication on Smartphones



vom Fachbereich Informatik
der Technischen Universität Darmstadt
genehmigte

DISSERTATION

zur Erlangung des akademischen Grades eines
Doktor-Ingenieurs (Dr.-Ing.)
von

Dipl.-Math. Claudia Nickel

geboren in Hildesheim, Deutschland

Referenten der Arbeit: Prof. Dr. Johannes Buchmann
Technische Universität Darmstadt

Prof. Dr. Christoph Busch
Hochschule Darmstadt

Tag der Einreichung: 19.03.2012
Tag der mündlichen Prüfung: 02.05.2012
Hochschulkennziffer: D 17

Darmstadt 2012

ERKLÄRUNG ZUR DISSERTATION

Hiermit versichere ich, die vorliegende Dissertation selbständig nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus Quellen entnommen wurden, sind als solche kenntlich gemacht. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Darmstadt, den 19. März 2012

Claudia Nickel

WISSENSCHAFTLICHER WERDEGANG

Januar 2009 - heute

Wissenschaftliche Mitarbeiterin an der Hochschule Darmstadt (CASED) und Promotionsstudentin an der Technischen Universität Darmstadt.

Oktober 2010 - Dezember 2010

Forschungsaufenthalt am Gjøvik University College, Norwegen.

Mai 2007 - Dezember 2008

Wissenschaftliche Mitarbeiterin am Fraunhofer-Institut für Graphische Datenverarbeitung IGD in Darmstadt.

Oktober 2002 - April 2007

Studium “Mathematics with Computer Science” an der Technischen Universität Darmstadt.

September 2004 - Juni 2005

Auslandsstudium an der Swansea University in Großbritannien.

First Author

1. C. Nickel, “Authentisierung an mobilen Geräten mittels Gangerkennung”, Datenschutz und Datensicherheit, 2009.
2. C. Nickel, X. Zhou, and C. Busch, “Template Protection via Piecewise Hashing”, IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009.
3. C. Nickel, X. Zhou, and C. Busch, “Template Protection for Biometric Gait Data”, BIOSIG 2010 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures, 2010.
4. C. Nickel, C. Busch, S. Rangarajan, and M. Möbius, “Using Hidden Markov Models for Accelerometer-based Biometric Gait Recognition”, 7th IEEE International Colloquium on Signal Processing & Its Applications, 2011 (*Best Paper Award*).
5. C. Nickel, M. O. Derawi, P. Bours, and C. Busch, “Scenario Test of Accelerometer-based Biometric Gait Recognition”, 3rd IEEE International Workshop on Security and Communication Networks, 2011.
6. C. Nickel, H. Brandt, and C. Busch, “Classification of Acceleration Data for Biometric Gait Recognition on Mobile Devices”, BIOSIG 2011 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures, 2011.
7. C. Nickel, and C. Busch, “Classifying Accelerometer Data via Hidden Markov Models to Authenticate People by the Way they Walk”, 45th IEEE International Carnahan Conference on Security Technology, 2011.
8. C. Nickel, H. Brandt, and C. Busch, “Benchmarking the Performance of SVMs and HMMs for Accelerometer-based Biometric Gait Recognition”, IEEE International Symposium on Signal Processing and Information Technology, 2011.

-
9. C. Nickel, T. Wirtl, and C. Busch, "Authentication of Smartphone Users Based on the Way they Walk Using k-NN Algorithm", 8th IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2012.
 10. C. Nickel, and C. Busch, "Does a Cycle-based Segmentation Improve Accelerometer-based Biometric Gait Recognition?", 11th IEEE International Conference on Information Science, Signal Processing and their Applications, 2012.

Co-Author

1. F. Breitinger and C. Nickel, "User Survey on Phone Security", BIOSIG 2010 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures, 2010.
2. H. Witte, and C. Nickel, "Modular Biometric Authentication Service System (MBASSy)", BIOSIG 2010 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures, 2010.
3. S. Abt, C. Busch, and C. Nickel, "Applikation des DBSCAN Clustering-Verfahrens zur Generierung von Ground-Truth Fingerabdruck-Minutien", BIOSIG 2010 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures, 2010.
4. M. O. Derawi, C. Nickel, P. Bours, and C. Busch, "Unobtrusive User-Authentication on Mobile Phones using Biometric Gait Recognition", 6th IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2010 (*Best Paper Award*).
5. C. Busch, S. Abt, C. Nickel, U. Korte, and X. Zhou, "Biometrische Template-Protection-Verfahren und Interoperabilitätsstrategien", Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V., 2010.
6. C. Busch, U. Korte, S. Abt, C. Böhm, I. Färber, S. Fries, J. Merkle, C. Nickel, A. Nouak, A. Opel, A. Oswald, T. Seidl, B. Wackersreuther, P. Wackersreuther, and X. Zhou, "Biometric Template Protection", Datenschutz und Datensicherheit, 2011.
7. C. Stein, and C. Nickel, "Eignung von Smartphone-Kameras zur Fingerabdruckerkennung und Methoden zur Verbesserung der Qualität der Fingerbilder", BIOSIG 2011 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures, 2011.
8. T. Wirtl, and C. Nickel, "Aktivitätserkennung auf Smartphones", BIOSIG 2011 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures, 2011.
9. M. Hestbek, C. Nickel, and C. Busch, "Biometric Gait Recognition for Mobile Devices Using Wavelet Transform and Support Vector Machines", 19th IEEE International Conference on Systems, Signals and Image Processing, 2012.
10. M. Muaaz, and C. Nickel, "Influence of Different Surfaces and Walking Speeds on Accelerometer-based Biometric Gait Recognition", 35th IEEE International Conference on Telecommunications and Signal Processing, 2012.

ACKNOWLEDGMENTS

First of all I would like to thank Christoph Busch for giving me the opportunity to write my PhD-thesis under his supervision. I really appreciate how much time you had for me. Thanks for the guidance and good advices. I would also like to thank Johannes Buchmann for accepting me as a PhD-student at the computer science department of Technische Universität Darmstadt.

I had the pleasure to be one of the first PhD-students starting at CASED and I really enjoyed the atmosphere during the last three years. Thanks to Michael Kreutzer for always having a smile for everybody. Very special thanks go to all members of CASED and NISlab participating in our data collections. I know it has been sometimes exhausting, but with your walk you built the foundation for my thesis. Mark, it has been fun sharing the office with you, thanks for all the help with various PC-problems.

Thanks to all the students who worked for me during the last three years. Especially Heiko, Holger, Tobias, Salman, Sathya, Martin H. and Leqiao did support me a lot. I would also like to thank Mohammad for the busy times we spent together in Darmstadt and Gjøvik. Thanks to Manuel who introduced me to the world of speaker recognition. Thanks to Xuebing for being a great colleague and friend and for steadily supplying me with pralines and mochi while I was writing the thesis. It was good to have all of you around to discuss with.

Although I wrote this thesis at CASED, everything started at Fraunhofer IGD. I would like to thank Wolfgang, Elke and Silke for initiating my scientific career at Fraunhofer IGD and letting it start with an unforgettable phone-call. Thanks to Uli for giving me the possibility for the internship and Hiwi-job. It was a great time during which I learned a lot. Thanks to Alexander for entrusting me with the Theseus project. Looking back, I am glad for all the experiences it gave me and all the nice people I met. Thanks to Alex and his coffee machine for making many afternoons more agreeable. Having been attached to the department for nearly seven years, there are several more people I have to thank for their support, like Wolfgang, Jan, Martin, Peter, Frank, and Helmut.

Thanks to Thomas for his love and support and patience, especially at the end of the thesis. Thanks go also to my family and my friends on which I can always rely.

The authentication via accelerometer-based biometric gait recognition offers a user-friendly alternative to common authentication methods on smartphones. It has the great advantage that the authentication can be performed without user interaction. When the user is walking, his walk-pattern can be extracted from the accelerations measured using the integrated sensors of the smartphone. This pattern can be used for authentication.

A study [24] showed that users often deactivate the authentication methods of their mobile devices because they consume too much time. Because all steps necessary to perform biometric gait recognition can be executed in the background, no user interaction is necessary for the presented technique. Performing a continuous authentication while the user is walking, an up-to-date authentication result is available at any point in time. During log-in, no calculations are necessary anymore, hence there is no delay. Only in cases where the user is not walking, an alternative authentication method has to be used.

This is a great benefit for the user because he has the advantages of a phone which is protected by authentication but without the disadvantages common methods impose. This high user-friendliness is likely to increase the number of smartphones for which the screen-lock is linked with an authentication. Therefore, a higher security of the data stored in smartphones can be achieved. A misuse of the stored information by an unauthorized user can be prevented.

Due to the growing distribution of powerful smartphones, the number of available applications is increasing as well. These applications result in a growing amount of data stored on the devices, which make the protection of the device necessary. These data comprise e. g. addresses, appointments or GPS-information. Additionally, some applications, e. g. of e-mail-providers or social networks, require the user to authenticate himself. Often these credentials are stored by the user on the phone, such that it is not necessary to enter them each time. In case an unauthorized person has access to such a phone he can use these services without restrictions and therefore substantially harm the user.

The objective of this thesis was to develop methods for accelerometer-based biometric gait recognition which achieve sufficient low error rates, as well as to demonstrate that their computational effort is low and allows for an execution on current smartphones. Because the basis of existing methods is the extraction of gait cycles (i. e. two steps) from the accelerometer data, a cycle-based method was developed and evaluated in a scenario test. This method uses raw data of the gait cycles as feature vectors and accomplishes the classification using distance functions.

In addition, a further approach was selected, which does not need the time-costly and error-prone gait cycle extraction. Instead, it is using overlapping segments of a fixed time length. Several features are extracted from these segments and combined to feature vectors. Machine learning algorithms are used for classification. A benchmark of the approaches on a challenging database showed that these methods yield low equal error rates between 6% and 7% and are outperforming the cycle-based methods. These error rates were achieved under the realistic conditions that training and probe data are not collected on the same day.

It was shown that five minutes of gait data are sufficient to thoroughly train the models. It should be regarded that the training data contain the different walking velocities at which the user should be recognized later on. To obtain low false rejection rates, the classification should be based on around three minutes walk data. Two of the developed methods were implemented on a smartphone. It was shown that both methods are able to perform the classification fast enough to allow for an authentication without delay for the user.

Die Authentisierung mittels beschleunigungsbasierter biometrischer Gangerkennung ist eine nutzerfreundliche Alternative zu herkömmlichen Authentisierungsverfahren für Smartphones. Diese hat den großen Vorteil, dass die Authentisierung ohne Nutzerinteraktion durchgeführt wird. Wenn der Nutzer geht, wird über die bereits werkseitig in das Telefon integrierten Beschleunigungssensoren sein spezifisches Gangmuster bestimmt und zur Authentisierung verwendet.

In einer Studie [24] wurde gezeigt, dass die Nutzer die Authentisierungsfunktionen ihres Handys häufig deaktivieren, da diese zu zeitaufwändig sind. Da alle für die Gangerkennung durchzuführenden Schritte im Hintergrund ablaufen, ist für das vorgestellte Verfahren keine Nutzerinteraktion notwendig. Durch eine fortlaufende Berechnung der Authentisierungsergebnisse während der Nutzer läuft, ist ein aktuelles Ergebnis jederzeit vorhanden. Zum Log-in Zeitpunkt sind daher keine Berechnungen mehr notwendig und es tritt somit auch kein Zeitverlust auf. Lediglich falls der Nutzer nicht geht, ist eine andere Authentisierungsmethode notwendig.

Dies ist ein erheblicher Nutzen für den Anwender, da er von den Vorteilen eines durch Authentisierung geschützten Smartphones profitieren kann, ohne jedoch die Nachteile herkömmlicher Verfahren in Kauf nehmen zu müssen. Diese hohe Nutzerfreundlichkeit kann dazu führen, dass die Anzahl der Smartphones steigt, bei denen die Bildschirmsperrung an eine Authentisierung gekoppelt ist. Somit kann der Sicherheitslevel der in den Smartphones gespeicherten Daten erhöht und ein Missbrauch dieser Informationen durch einen unberechtigten Nutzer verhindert werden.

Durch die wachsende Verbreitung leistungsfähiger Smartphones steigt auch die Anzahl zur Verfügung stehender Anwendungen stetig. Diese Anwendungen führen dazu, dass immer mehr Daten auf den Geräten gespeichert werden, die einen Schutz des Gerätes notwendig machen. Dies können z.B. Adressen, Termine oder GPS-Daten sein. Außerdem erfordern einige dieser Applikationen, z.B. von sozialen Netzwerken oder E-Mail-Anbietern, dass sich der Nutzer authentisiert. Häufig werden diese Zugangsdaten gespeichert, so dass sie nicht bei jeder Nutzung eingegeben werden müssen. Hat ein unberechtigter Nutzer Zugriff auf das Smartphone kann er diese Dienste uneingeschränkt verwenden und somit dem Eigentümer erheblichen Schaden zufügen.

Ziel dieser Arbeit war, Verfahren für die beschleunigungsbasierte biometrische Gangerkennung zu entwickeln, die ausreichend niedrige Fehlerraten bieten und zu zeigen, dass diese leistungsfähig genug sind, um auf aktuellen Smartphones ausgeführt zu werden. Da die Grundlage bestehender Verfahren die Extraktion von Gangzyklen (zwei Schritten) aus den Beschleunigungsdaten ist, wurde ein zyklusbasiertes Verfahren entwickelt und in einem Szenariotest evaluiert. Dieses Verfahren

verwendet die Rohdaten der Gangzyklen als Merkmalsvektoren und führt eine Klassifizierung mit Hilfe von Distanzfunktionen durch.

Desweiteren wurde ein Ansatz gewählt, der ohne die aufwändige und fehlerbehaftete Extraktion der Gangzyklen auskommt und stattdessen eine Segmentierung in überlappende Segmente fester Länge nutzt. Aus diesen Segmenten wurden verschiedene Merkmale extrahiert und zu Merkmalsvektoren kombiniert. Die Klassifizierung wurde mittels Verfahren des maschinellen Lernens durchgeführt. Ein Vergleich der Verfahren auf einer anspruchsvollen Datenbasis zeigte, dass diese Verfahren niedrige Gleichfehlerraten zwischen 6% und 7% erreichen und damit besser sind als das zyklusbasierte Verfahren. Diese Fehlerraten wurden unter der realistischen Bedingung erreicht, dass Trainings- und Probedaten nicht an dem selben Tag aufgenommen wurden.

Es wurde gezeigt, dass fünf Minuten Gangdaten genügen, um die Modelle ausreichend zu trainieren. Hierbei sollte beachtet werden, dass die Trainingsdaten die verschiedenen Geschwindigkeiten enthalten bei denen der Nutzer später erkannt werden soll. Um niedrige Falschrückweisungsrate zu erhalten, sollte die Klassifizierung auf circa drei Minuten Gangdaten basieren. Zwei der entwickelten Verfahren wurden für Smartphones implementiert und evaluiert. Es wurde gezeigt, dass die Klassifizierung bei beiden Verfahren schnell genug durchgeführt werden kann, so dass es möglich ist, eine Authentisierung ohne Zeitverlust für den Nutzer durchzuführen.

1. Introduction	1
1.1. Research Questions	2
1.2. Structure of the Thesis	3
2. Authentication on Smartphones	5
2.1. PIN or Password	5
2.2. Graphical Authentication Methods	6
2.2.1. Locimetric Methods	6
2.2.2. Drawmetric Methods	6
2.2.3. Cognometric Methods	7
2.3. Biometrics	7
2.3.1. Face Recognition	8
2.3.2. Speaker Recognition	8
2.3.3. Fingerprint Recognition	9
2.3.4. Existing Products	9
2.4. Survey on Phone Usage and Security	10
2.4.1. Phone Familiarity, Usage, and Knowledge of IT-Security	11
2.4.2. Used Phone Features and Stored Data	11
2.4.3. Carrying, Attending the Phone and Phone Theft	11
2.4.4. Phone Security	12
2.4.5. Using Biometrics	12
2.5. Summary	14
3. State of the Art of Biometric Gait Recognition	15
3.1. Gait	15
3.2. Machine-Vision-based Biometric Gait Recognition	16
3.2.1. 3D Videos	18
3.2.2. Surveillance	18
3.2.3. Influence of Covariates	19
3.3. Floor-Sensor-based Biometric Gait Recognition	19

3.4. Wearable-Sensor-based Biometric Gait Recognition	20
3.4.1. VTT Technical Research Centre of Finland	21
3.4.2. Norwegian Information Security laboratory at Gjøvik University College . .	21
3.4.3. Further Research	24
3.5. Summary	26
4. Design of an Accelerometer-based Biometric Gait Recognition System	27
4.1. Modular Biometric Authentication Service System	27
4.1.1. Database	28
4.1.2. Modules	29
4.1.3. User Interface	29
4.1.4. Background Service	29
4.2. Components of a General Biometric System	29
4.2.1. Data Capture Subsystem	29
4.2.2. Signal Processing Subsystem	31
4.2.3. Data Storage Subsystem	31
4.2.4. Comparison Subsystem	31
4.2.5. Decision Subsystem	32
4.3. Functions of the Modular Biometric Authentication Service System	32
4.3.1. Enrolment	32
4.3.2. Authentication	32
4.4. Summary	34
5. Feature Extraction and Classification	35
5.1. Preprocessing	35
5.2. Segmentation	38
5.2.1. Cycle-based Segmentation	38
5.2.2. Fixed-length Time Segmentation	40
5.3. Feature Extraction	40
5.3.1. Mel- and Bark-Frequency Cepstral Coefficients and Variations	41
5.3.2. Feature Selection Using Discriminative Potential Score	42
5.4. Classification	43
5.4.1. Template-based Classification	43
5.4.2. Machine Learning Classification	44
5.5. Voting	49
5.6. Summary	50
6. Benchmarking Different Accelerometer-based Biometric Gait Recognition Methods	51
6.1. Initial Data Collection Using a G1 Phone	51
6.2. Hidden Markov Models	52
6.2.1. Evaluation	53
6.3. Support Vector Machines	54
6.3.1. Evaluation	55
6.4. Benchmark of the Approaches	58
6.5. Summary and Conclusions	59
7. Evaluation on a Large Database	61

7.1. Longer Data Collection Using Motorola Milestone	61
7.2. Hidden Markov Models	62
7.2.1. Evaluation	62
7.3. Support Vector Machines	70
7.3.1. Evaluation	70
7.4. k-Nearest Neighbor Algorithm	77
7.4.1. Evaluation	77
7.4.2. Implementation as MBASSy Module	82
7.5. Combining the Cycle-based Segmentation with Classification via HMMs	84
7.5.1. Varying the Sampling Rate	84
7.5.2. Varying the Normalized Cycle Length	85
7.5.3. Concatenating Cycles	85
7.5.4. Comparison of Different Features	86
7.5.5. Voting	87
7.5.6. Benchmarking Cycle-based against Fixed-length Time Segmentation	87
7.6. Benchmark with Published State-of-the-Art Method	88
7.6.1. Data Preparation	88
7.6.2. Evaluation	89
7.6.3. Benchmark to Frank's Results	89
7.6.4. Benchmark to Results of this Chapter	90
7.7. Summary and Conclusions	90
8. Challenging Evaluation on a Realistic Database	95
8.1. Challenging Data Collection Using Motorola Milestone	95
8.2. Hidden Markov Models	98
8.2.1. Evaluation	98
8.3. Support Vector Machines	102
8.3.1. Evaluation	102
8.4. k-Nearest Neighbor Algorithm	108
8.4.1. Evaluation	108
8.5. Cycle-based Method	111
8.5.1. Evaluation	114
8.6. Summary and Conclusions	116
9. Conclusions and Future Work	119
9.1. Answers to Research Questions	119
9.2. Benchmark of the Algorithms	122
9.3. Simulation Study	122
9.4. Future Work	123
A. Survey	125
B. HMM Results for Database 2	127
C. SVM Results for Database 2	131
D. k-NN Results for Database 2	137

E. HMM Results for Database 3	141
F. SVM Results for Database 3	147
G. k-NN Results for Database 3	153
H. Feature Sets Based on Discriminative Potential Score	157
Glossary	161
Bibliography	165

The focus of this thesis is the unobtrusive authentication on smartphones via accelerometer-based biometric gait recognition. Such an authentication method enables the mobile phone to recognize its owner based on the way he is walking. There are two main advantages of this approach. First, gait can be captured via acceleration sensors, which are already integrated in smartphones. Hence, there are no additional hardware costs for deploying this method. Second, gait recognition does not require explicit user interaction during verification as the phone does it literally “on-the-go”. These two factors assure the high usability of accelerometer-based biometric gait recognition, which does not require extra interaction time.

This overcomes the problems of conventional authentication methods on mobile devices. Currently, most phones offer only authentication via Personal Identification Number (PIN). Recently, graphical authentication techniques became more adapted [124]. Section 2.4 presents results of a survey, showing that most owners of mobile devices do not activate the PIN¹ authentication. The main reason for this is the low user-friendliness because of a high time-consumption.

Current statistics further emphasize the relevance of this topic. The number of mobile phone subscribers increased to more than five billion in 2010 [72] and nearly 70% of the consumers are using smartphones [117]. In addition, the computational power of smartphones is steadily increasing, resulting in a wide variety of different applications. Some of these applications, like online banking, require the user to authenticate himself. Other applications store various kinds of data in the phone. Because these data often contain sensitive private or business information, the phone itself should be protected by an authentication method. Mobile phones can be easily lost or forgotten and they are a facile target for thefts. When no authentication is required, the attacker can effortlessly analyze the data on the phone or use stored passwords to access e-mail or social network accounts.

Different authentication methods exist. The most common one is the knowledge-based authentication, where the user has to enter a secret, e.g. a PIN or password. Token-based authentication methods require the user to present some kind of physical object like a smartcard or a regular door key. This thesis focuses on biometric authentication. Biometric methods authenticate the user

¹All acronyms are explained in the Glossary.

based on his physiological or behavioral characteristics². The main advantage of these methods is that the secret necessary for authentication is directly linked to the subject; hence, it cannot be forgotten or passed to somebody else easily.

The main procedure when using a biometric system is as follows. During enrolment, the subject presents his biometric characteristic to the sensor and the biometric reference is computed and stored for later comparison. During authentication, the biometric characteristic is captured again and the extracted biometric probe is compared to the reference. Based on the similarity or dissimilarity of the two data, the subject is authenticated or rejected. Various systems using different biometric modalities exist; the most common is fingerprint authentication.

In this thesis a behavioral biometric modality is evaluated, namely gait, which is the way a subject is walking. The data are captured via accelerometers contained in smartphones. Research on accelerometer-based biometric gait recognition was started by Ailisto et al. resulting in the first publication on this topic in 2005 [6]. The technique was further developed and analyzed by Gafurov [50]. In these initial publications, dedicated prototype accelerometers were used, which were attached to the hip, leg, arm or ankle of the subjects.

Previous research on accelerometer-based gait recognition indicated that this biometric modality has promising features. Nevertheless, the so far conducted research had several drawbacks.

- The evaluation databases were collected using dedicated sensors; hence it was unclear if the built-in sensors of mobile phones are of suitable precision.
- The main focus was on extracting gait cycles and applying template-based classification algorithms. Other approaches, like machine learning algorithms, have not been considered although they showed promising results for different biometric modalities.
- The testing databases do mainly contain only walking straight on flat floor and often the data are collected on one single day.

Solving these drawbacks is the motivation to the research described in this thesis. This is reflected by the research questions presented in the following section.

1.1. Research Questions

The main research question motivating the work in this thesis is

“Can accelerometer-based biometric gait recognition be used as an authentication method on smartphones?”.

The before-mentioned drawbacks of previous research have to be overcome to answer this question. The research is divided into the following more specific questions:

1. Are machine learning algorithms more suitable than the formerly proposed gait recognition methods using gait cycles and template-based comparisons?
2. What are suitable features to describe subject specific gait?
3. How much training data are necessary when applying machine learning algorithms to accelerometer-based biometric gait recognition?

²The biometric terms are used in accordance to the ISO SC37 Harmonized Biometric Vocabulary (<http://www.3dface.org/media/vocabulary.html>).

4. Does the biometric performance decrease with aging of the reference data?
5. How is the biometric performance when reference and probe are obtained from data collected with different walk velocities?
6. How is the biometric performance of the proposed methods in a scenario test?
7. Are current smartphones capable of performing the necessary feature extraction and classification steps with an acceptable computational complexity?

1.2. Structure of the Thesis

This thesis considers the aspects to be solved in order to deploy accelerometer-based biometric gait recognition as an authentication method on smartphones. To introduce the reader to the area of authentication on mobile phones and to gait recognition, the state of the art for these topics is presented. The design of an application for authentication on mobile devices is presented. The focus of the thesis is the identification of relevant features and composition of suitable gait recognition methods. Different databases focusing on different goals are applied in the evaluation. To get comparable results, several methods are evaluated in a similar manner on each database. The following sections give a short overview over the remaining part of the thesis.

Chapter 2: Authentication on Smartphones

For authentication on mobile phones, two different approaches are relevant. The most popular is PIN authentication, which is a knowledge-based authentication method. Graphical authentication methods belong to the same group. These methods help the user to choose and remember the secret key by providing some kind of graphical assistance. Biometric authentication is an alternative approach with the advantage that the user does not have to remember the secret key because it is extracted from the required biometric characteristic. Various methods for both approaches, knowledge-based and biometrics, are presented in this chapter. In addition, the results of a user survey concerning authentication on mobile devices are presented.

Chapter 3: State of the Art of Biometric Gait Recognition

The way a subject is walking can be recorded via different kinds of sensors. Chapter 3 presents the different approaches. The most established approach is to use cameras as they are installed for surveillance purpose, but also floor sensors are possible. Although these two approaches are not relevant for authentication on mobile phones, they underline the suitability of gait as biometric modality. Starting only in 2005, accelerometer-based biometric gait recognition is still a very new field of research. This allows to present all relevant publications in this chapter. The indicated shortcomings of the state of the art demonstrate the need for the research presented in this thesis.

Chapter 4: Design of an Accelerometer-based Biometric Gait Recognition System

This chapter presents the design of a modular biometric authentication service system for Android OS phones. The system offers the possibility to integrate and test several authentication methods,

like gait recognition or graphical authentication. Referring to this system, the components of a general biometric system as given in ISO/IEC SC37 Standing Document 11 are explained. The three function modes, namely enrolment, verification, and identification are described.

Chapter 5: Feature Extraction and Classification

The developed gait recognition components are presented in this chapter. Feature extraction requires either the extraction of so-called gait cycles, containing the accelerations measured during two steps, or the data are divided into segments of fixed length. Several features are described which can be extracted from the segments. For classification of gait cycles, template-based comparison methods are usually applied. For classification of non-cycle-based feature vectors, machine learning algorithms are proposed. The evaluated classification methods are presented in this chapter.

Chapter 6: Benchmarking Different Accelerometer-based Biometric Gait Recognition Methods

In total three different data collections were composed and are used for evaluation in this thesis. This chapter states the results for the initial data collection using the Google G1 [64] phone. Fixed-length time segments were extracted from the data and machine learning algorithms were used for classification. Hidden Markov Models (HMMs) were applied to classify the raw data. Feature vectors were created for classification with Support Vector Machines (SVMs). A comparison of the error rates to the ones obtained when applying a cycle extraction technique to the same database is conducted.

Chapter 7: Evaluation on a Large Database

Due to the fast developments on the mobile phone market during the work on this thesis, soon further Android phones were available which had better processors and made it possible to get acceleration values at a higher rate. One goal of the first data collection using a new phone was to get more data for each subject, which was necessary to deeply test the methods using machine learning algorithms. SVMs, HMMs, and the k-Nearest Neighbor (k-NN) algorithm are thoroughly evaluated and suitable preprocessing configurations as well as features are determined. The influence of the enrolment duration and different walk velocities is analyzed.

Chapter 8: Challenging Evaluation on a Realistic Database

The third data collection had the goal to perform a scenario test on a close to realistic data set. This means that subjects did not only walk up and down the hall on flat floor, but also came across doors, stairs, chairs etc. This chapter presents the results for SVMs, HMMs, the k-NN algorithm, and the developed cycle extraction technique. The thorough evaluation presents amongst others the best amount of training data, and classification results for each subject.

Chapter 9: Conclusions and Future Work

Based on the results described in the previous chapters, final conclusions are made in this chapter. The research questions and respective answers are summarized. In addition, proposals for future work are made.

CHAPTER 2

AUTHENTICATION ON SMARTPHONES

The development of cell phones has progressed rapidly during the last two decades. Initially, they were bulky devices, only allowing to make phone calls or write short messages (SMS), and they were mainly used by business-men. Nowadays, nearly everybody uses cell phones or even smartphones. These are constantly gaining more and more functionality, and the number of available applications is steadily increasing. As a result, the amount and variety of stored data also increases. An impostor gaining access to phone numbers, calendar-entries, e-mails, social network accounts etc. can impersonate and harm the owner of the phone. Therefore, it is important to secure the phone and the data stored on it by an authentication method. But the survey described in Section 2.4 shows that for most mobile devices, entering a PIN is not required when reactivating it after a stand-by phase, allowing attackers easy access to the data.

So far, most phones only offer a knowledge-based authentication via PIN or password, and these methods are described in Section 2.1. More user-friendly alternatives are graphical authentication methods (Section 2.2) and biometrics (Section 2.3), which is the focus of this thesis. In addition to using these methods for authentication for accessing the mobile device, they can also be used for secure access to services.

2.1. PIN or Password

Knowledge-based authentication via PIN or password is the most common method. In the context of mobile devices, a PIN is often used for authentication at the Subscriber Identity Module (SIM)-card after starting the phone. However, this does not protect the content of the phone storage as it can be circumvented by removing the SIM-card from the phone.

The main draw-back of knowledge-based authentication is that the required secret is either secure or easy-to-remember. Users have to choose different passwords for a huge number of applications, and they are not able to remember a secure PIN/password for each of them. This results in short PINs/passwords, or ones which contain semantic information like birthdays, pet names etc. As a result, the search space for an attacker is decreased and dictionary attacks [96] are likely to succeed.

2.2. Graphical Authentication Methods

The motivation behind using graphical authentication methods is twofold. First, a higher level of security is obtained because the secret needed for authentication is either an image or a pattern. This is harder to write down or to pass to somebody else than a conventional password. Second, it is assumed that the secret can be easier recalled because according to the so-called *picture superiority effect*, pictures are easier to remember than words [113].

De Angeli et al. [37] categorize graphical authentication methods into three groups: *locimetrics*, *drawmetrics*, and *cognometrics*. For Locimetric methods it is necessary to identify special points in one single image. Drawmetric methods request the user to reproduce a previously drawn image. Cognometric methods require the user to recognize previously selected images in a set of distractor images. These three categories are described in more detail in the following subsections. Sample images for each of the three approaches are given in Figure 2.1.

2.2.1. Locimetric Methods

In 1996, Blonder registered a patent called *Graphical Password* [18]. During enrolment, the user taps on freely choosable locations on a given image. These chosen locations form the password and are stored in the system. During authentication, the user needs to tap on the same locations again. The system may specify the required accuracy of the taps, as well as the number of correct taps needed for authentication. The disadvantage of this system is the limited number of specific points available in an image [154]. As a result, different users tend to choose the same locations as password. An implementation of this approach (called PassPoints) is proposed by Wiedenbeck [155].

One commercial application exists based on this approach: Visual Key [129] by SFR Software GmbH. Until end of 2010, a similar product called V-go Password was offered by Passlogix. After its take-over by Oracle, the continuation of V-go Password is uncertain.

2.2.2. Drawmetric Methods

The best-known drawmetric method is *Draw-A-Secret (DAS)*, which was proposed by Jermyn et al. in 1999 [75]. In contrast to the *Graphical Passwords*, where the location tapped during authentication only needs to be close to the one tapped during enrolment, DAS requires the secret to be exactly repeated.

The display is divided into a rectangular grid. During enrolment, the user draws a pattern on that grid. The coordinates of each passed cell are stored in the order in which they were crossed. In addition, each pen-up event (end of one stroke) is stored. For authentication the same pattern has to be repeated. The required accuracy depends on the chosen cell size. The security depends on the length and number of strokes which form the secret.

Thorpe et al. [141] enhanced the security of the system by proposing a grid selection. First, the user has to select a target region in a large grid. Afterwards, he has to draw his pattern on a zoomed-in version of that region, adding an additional complexity to guessing the secret.

The DAS-approach is also used by the graphical authentication offered on Android phones (*lock screen pattern*). Here, the user has to connect dots which are displayed in a 3×3 grid. In 2011, Umar and Rafiq [144] proposed an advanced drawmetric method where the user can draw geometrical shapes consisting of lines and triangles. This results in a high secret space while maintaining a high usability.

2.2.3. Cognometric Methods

Most published methods belong to the group of cognometric methods. In each case, authentication is only successful if the user chooses the same images as during enrolment. The methods only differ with respect to how many images are displayed, how they are arranged (e.g. on one or multiple screens), and which kind of images have to be remembered. These might be faces (Passfaces [2]), cartoon characters (ToonPasswords [68]), general photographic images (Awase-E [140], Picture Password [74]), or abstract structured images (Déja Vu [43]). The security depends on the number of images the user has to choose and the number of images which are displayed. Davis et al. [36] analyzed the security of Passfaces. They showed that the selection of faces greatly depends on the user's sex and race, which made it easy for an attacker to choose the correct faces when aware of this additional information.

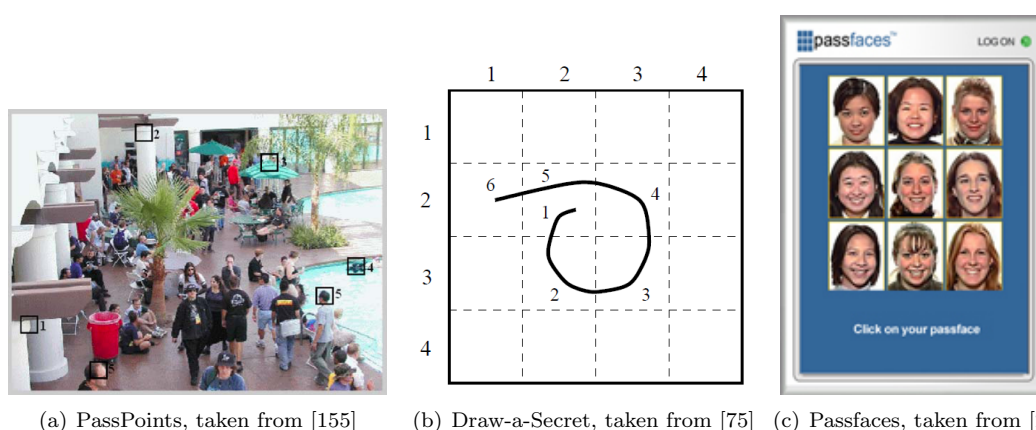


Figure 2.1.: Sample images for the three different approaches of graphical authentication: 2.1(a) locimetric, 2.1(b) drawmetric, 2.1(c) cognometric.

2.3. Biometrics

Biometrics can be used to authenticate individuals based on their behavioral or physiological characteristics. Research has been done on different modalities like fingerprint, face, iris, voice, gait, and keystroke. With the increasing computational power of mobile devices, it is possible to integrate these methods for authentication. In 2011, Goode Intelligence published a report in which they forecast that the mobile phone biometric security market will grow from \$30 million in 2011 to over \$161 million in 2015.

In the following, more details on three of the currently most relevant modalities for mobile phones will be given, namely face recognition, speaker recognition, and fingerprint recognition. This is not a complete presentation, but it should give an impression of the current work in the respective areas. All described methods aim at running completely on the phone. As gait recognition is in the focus of this thesis, it will be covered in-depth in the next chapter. Further biometric methods proposed for authentication on mobile devices include iris [123], signature [89], hand geometry [38], and in-air signatures [65].

2.3.1. Face Recognition

With the integration of front cameras in mobile phones, face recognition is becoming a very convenient mobile authentication method. As part of the MOBIO project [1], different face and speaker recognition systems were evaluated using a database collected using a mobile phone and a laptop [87]. 18 systems from nine institutes have been analyzed. The best result of a Half Total Error Rate (HTER) of 10.91 was obtained by a system fusing results based on Local Binary Pattern Histograms [30] with those of Multiscale Local Phase Quantization Histograms [31] and applying score normalization.

General research on face recognition is focused on decreasing error rates, e.g. by obtaining a higher robustness against changes in illumination or spoof detection. In the area of mobile devices, the main concerns are the restricted memory and computational power available, as well as the uncontrolled environment. Xi et al. [158] reduce the necessary resources by computing correlation filters, not on the full-size images, but on downsampled images and selected sub-regions (eye, nose etc.). The algorithm was implemented on the phone and it was shown that it consumes 1/30 time, 1/10 memory and 1/6 storage space compared to the conventional correlation method. Unfortunately, the algorithm was only tested using a conventional database which was not collected using a mobile phone, but it yielded a promising Equal Error Rate (EER) of 1.2%.

Because Eigenfaces [143] and Fisherfaces [14] are the most common face-recognition algorithms, Villegas et al. [149] optimized those for application on mobile devices. A memory reduction was achieved by using only 8-bit signed integers, and the computational time was reduced by applying mainly integer arithmetic. It was shown that these modifications do not significantly influence the error rates.

Ng et al. [104] propose to apply the Viola and Jones algorithm [150] for face detection. The face recognition uses unconstrained MACE filters [86], which are compared using peak-to-sidelobe ratio in order to improve robustness against different illuminations. Fixed point arithmetic is used to improve the performance of the algorithms.

2.3.2. Speaker Recognition

Speaker recognition on mobile devices can either be realized as continuous or explicit authentication. For a continuous authentication the speech during phone calls is analyzed and the authentication is performed in the background [78]. This can be used to directly lock the phone or to adjust the trust level [35]. The explicit authentication requires the user either to speak freely or recite given text when an authentication is requested. A further application of speaker recognition is memory assistance [85], preventing people from forgetting the name of the person they are talking to.

In 2011, Roy et al. [127] tested their previously proposed new speaker recognition method which uses Boosted Slice Classifiers on a database containing mobile phone data. They obtained an HTER of around 17%. Compared to the results of 17 other speaker recognition algorithms (described in [87]) for the same database (which range from HTER = 10.3% to 23.0%), this is one of the worse results. But comparing the computational complexity of the systems (measured by the number of floating point operations), the algorithm of Roy et al. greatly outperforms the others, showing its suitability for mobile devices.

By using HMMs for classification and Mel Frequency Cepstral Coefficients (MFCCs) as features, Kunz et al. [78] obtained an EER of 15%. Segments of two seconds were classified and the decision was based on the mean of six classification results. The system was implemented on a mobile device showing the feasibility of authenticating people in real-time.

2.3.3. Fingerprint Recognition

Two different realizations of fingerprint recognition on mobile phones exist. One possibility is to use a dedicated sensor. Although mobile phones with integrated fingerprint scanners are popular in Japan, they are not widespread in the rest of the world. The second approach is to use the built-in camera to capture an image of the finger (see Figure 2.2).

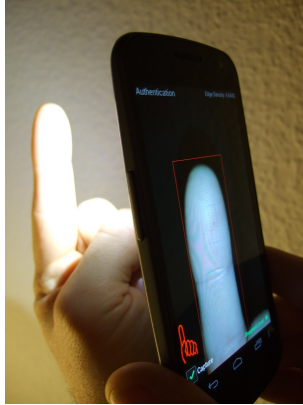


Figure 2.2.: Fingerprint recognition using the integrated camera of a smartphone.

Only few existing publications use the built-in camera of the phone. Stein and Nickel [134] report that several up-to-date phone cameras (like the one in the iPhone 4) were not able to focus on the finger, which disqualified them for use for fingerprint recognition. Only one phone could be used for the evaluation, but in most cases the images were of too low quality for fingerprint recognition. Derawi et al. [42] compared the results of two different phone models in two different settings. One was fixed on a stand which assures a constant distance to the captured fingers. The other one was held by an operator who took the photos. The error rates obtained using the free captured images were at least twice the ones obtained with the fixed setting. Due to the fact that different phone models were used, it is hard to say how much these results are influenced by the different hard- and software. But the results of both stated publications indicate that capturing high-quality finger photos which contain enough information for fingerprint recognition is still a problem for current mobile phones.

Lee et al. [81] also used a fixed camera setting and additional Light Emitting Diode (LED) lighting. Their research focused on segmentation and orientation estimation. Su et al. [137] proposed a fingerprint authentication system for mobile phones. The authentication is performed on the phone but an external sweep sensor is used for capturing the fingerprints. An EER of 4.13% was reported based on 480 fingerprints of 20 subjects.

2.3.4. Existing Products

Several different companies offer applications for biometric authentication on smartphones, although it is not clear from their webpages if the techniques already reached product status. In 2005, OMRON announced the “World’s First Face Recognition Technology for Mobile Phones” [115]. Mobbeel [95] offers solutions for iris, voice, and signature recognition on mobile devices. The idea of Biometry.com is to extract face, lip, and speech features while the user is speaking random

displayed numbers. When this implicit authentication is successful, the program notes a high trust level for that user, which may allow him to use the phone for large financial transactions. The trust level decreases over time. Each time the user is making a phone call his voice is analyzed and the trust level is adjusted [17].

ClassifEye [33] offers a fingerprint recognition system using the mobile phone camera. They state to be able to compensate for varying light conditions and finger positions. Pantech was the first manufacturer to put a fingerprint scanning technology in its GI 100 mobile phone in 2005 [121]. Only in 2011, the first Android phone (Motorola Atrix [98]) with integrated fingerprint sensor was put on the market. The corresponding SDK will allow the large group of Android developers to incorporate fingerprint authentication into their applications. The Google Galaxy Nexus (using Android 4.0) which has been on the market since November 2011 offers face recognition as alternative authentication method. The common disadvantage of the presented products is that none of them offers the required usability in terms of unobtrusively capturing biometric traits.

2.4. Survey on Phone Usage and Security

A survey was developed to get an impression on how people use their mobile devices and which security settings they choose. The survey was conducted three times, twice in Germany (2010 and 2011) and once in Denmark (2010). The results of the first German survey were already published in [24]. This section gives a combined evaluation of all three surveys, which are described in the following:

Survey 1 This survey was conducted in German. 548 users participated. 89% filled in the online survey, the others filled in the paper version, which was distributed at a gym and a physiotherapist.

Survey 2 198 people took part in this survey of whom 110 filled in the Danish version and 88 filled in the English version. Both were made available online.

Survey 3 This survey was made available online and in German. 216 people participated.

In total 962 people participated. The gender-distribution has been well balanced with 52.5% male and 46.5% female participants (1% did not provide information). Detailed age and gender distribution are given in Table 2.1.

	< 18	18-23	24-30	31-40	41-50	51-60	> 60	unknown	sum
male	48	137	177	57	36	31	18	1	505
female	42	88	148	53	47	37	31	1	447
unknown	1	1	0	0	2	0	0	6	10
total	91	226	325	110	85	68	49	8	962

Table 2.1.: Age and gender distribution of all participants.

There have been slight variations in the questions of the three surveys, which made it necessary to adapt the results before combining them. A baseline questionnaire is given in Appendix A. The necessary conversions are described in the sections covering the respective questions.

Question	1	2	3	4	no answ.
Importance of mobile device (1 = very important, 4 = unimportant)	35.0	41.3	19.6	3.4	0.6
Knowledge of the features of the device (1 = very good, 4 = no interest)	36.9	41.7	16.5	4.4	0.5
Knowledge of IT security (1 = very good, 4 = no interest)	18.3	33.7	34.5	12.5	1.0

Table 2.2.: General questions about familiarity and usage of mobile devices (in %).

2.4.1. Phone Familiarity, Usage, and Knowledge of IT-Security

The goal of the first four questions was to get an impression about the participant's security awareness and how he is using his phone. 67% of the participants indicated that they use their phone only privately, 4% use their phone only for business and 29% for both. The participants were requested to give a ranking between 1 and 4. Questions, results and meaning of the ranks are given in Table 2.2. In Survey 2 people could choose between five ranks. Half of the selections of the middle rank were assigned to rank 2 and the other half to rank 3 to allow for a unified evaluation.

2.4.2. Used Phone Features and Stored Data

The participants were asked which additional phone features they use and which data they store on the phone. Five possible phone features were provided and there was the possibility to add further ones. 92.9% of the users use SMS, 62.5% the camera, 53.3% use the calendar, 30.2% internet, and 22.6% e-mail. 16.8% entered further features like alarm clock or MP3-player.

For the question about the kind of data stored on the phone, it was possible to choose one or more of seven given data types and again people could add further ones. 98.8% store phone numbers, 44.4% appointments, 33.7% addresses, 26.4% e-mails, 23.1% photos¹, 16.6% birthdays, and 11.3% passwords/PINs. 1.8% gave additional answers, like music or notes.

2.4.3. Carrying, Attending the Phone and Phone Theft

Relevant for the area of gait recognition on smartphones is the question where subjects mainly carry their phones. Participants could select between back trouser pocket, front trouser pocket, breast pocket, belt pouch, backpack, purse, and jacket or give a free answer. The possibility *jacket* was not given in Survey 1. Ten participants of that survey entered *jacket* in the free answer field. It is possible that more people would have chosen it if it had been provided as answer. The answers vary greatly between men and women, see Figure 2.3 for details.

A further question was where people keep their phone. Is it always within reach, often unattended or even sometimes lent to somebody else? Most people (69%) do have their phone always close to them but 23% do leave their phone often unattended. 2% lend it sometimes to somebody else.

¹This option has not been available at Survey 1; otherwise this value would probably be higher. Only a few of the participants of Survey 1 entered *photos* as a free answer, although 65% stated that they use the camera.

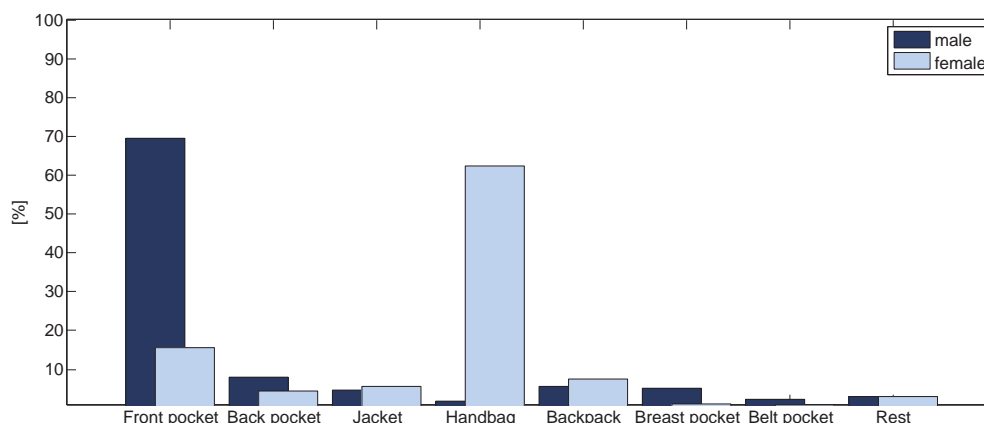


Figure 2.3.: Positions where the participants mainly carry their phones (in %).

12.3% of the participants had their phone lost or stolen once, additional 3.8% even more than once. In 16.8% this happened in public transport, in 4.0% in the city center, in 1.0% in the office and in 19.3% in a crowd of people. 58.9% used the possibility to give a free answer, like *school* or *at training*.

2.4.4. Phone Security

The results above show that a large amount of people store private information like e-mails or even PINs on the phone and many of them have been a victim of phone theft. So it is interesting to see how many of the phones are secured by a PIN to prevent the thief to be able to e.g. access e-mail accounts and further data. Figure 2.4 shows that only 18% use the PIN or the visual pattern provided on Android phones². People who are not using a PIN or visual pattern were asked for the reason. Figure 2.5 shows the possible answers. The main reason is that entering a PIN does take too much time. Many people (24.0%) did also state that they did not think about it. 36.7% of those stated that they have a very good or good knowledge of IT-security. This shows a low awareness of people regarding the necessity to protect cell phones. According to the answers many phones do not offer the PIN-authentication after stand-by. 19.0% chose this answer, and most of these (80.0%) stated in the beginning that they are very familiar or familiar with their phone settings. The age distribution in Figure 2.6 shows that especially the phones of younger and older people are not protected after a stand-by phase.

2.4.5. Using Biometrics

People were asked if they would use biometric authentication if their phone offered such. They were also requested to select which biometric modalities they would use. Results are given in Figure 2.7. 55.6% would use biometrics, and fingerprint is the favorite modality (89.9%). Gait recognition was only chosen by 8.6%, which may also be a result of the fact that this modality is

²This question was asked differently in the Danish version of Survey 2. Therefore, these participants are not considered for this question.

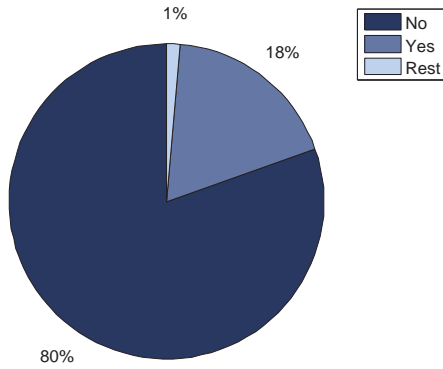


Figure 2.4.: Is the phone secured by PIN or visual pattern after a stand-by phase?

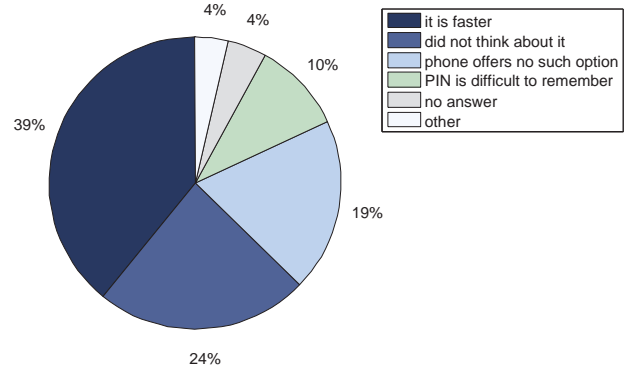


Figure 2.5.: If it is not secured, why did you use this setting?

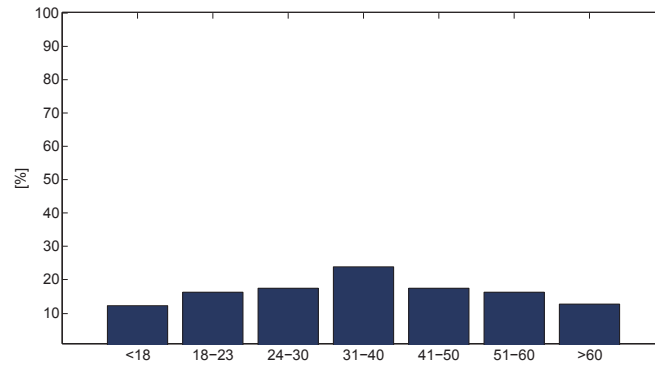


Figure 2.6.: Percentage of secured phones in each age group.

still unknown to most people and no additional information was given in the surveys. Considering only the people which currently do not protect their phone after a stand-by phase, 43.7% would use biometrics. This shows that offering biometric authentication for mobile devices would significantly increase the number of protected phones.

Figure 2.8 shows the willingness to use biometrics for the different age groups. It shows that especially young people are interested in applying biometric methods (76.9%). At the same time, this is the group which currently has the lowest percentage of secured phones (12.1%, see Figure 2.6). Assuming that all participants which are willing to use biometrics would use such a method for authentication after a stand-by phase, a high number of secured phones would be achieved. Figure 2.9 shows for each age group the percentage of participants which either are already using an authentication at the moment or are currently not using an authentication but are willing to use biometric authentication if provided. Comparing Figure 2.9 and 2.6, one can see huge increase in secured phones.

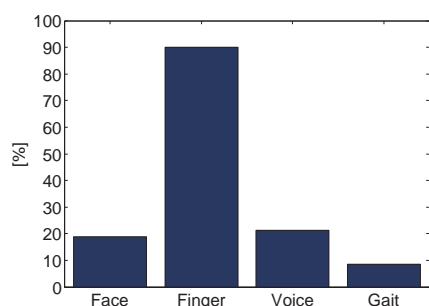


Figure 2.7.: The four possible choices for biometric modalities. Bars state the percentages of participants with interest in biometrics which would use that specific modality.

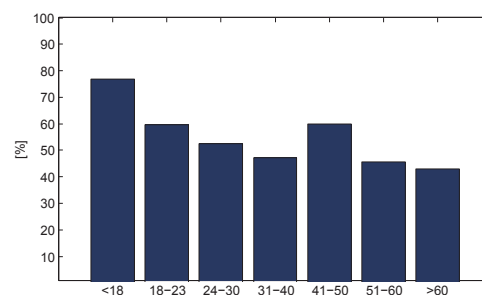


Figure 2.8.: Willingness to use biometrics (percentage of the specified age group).

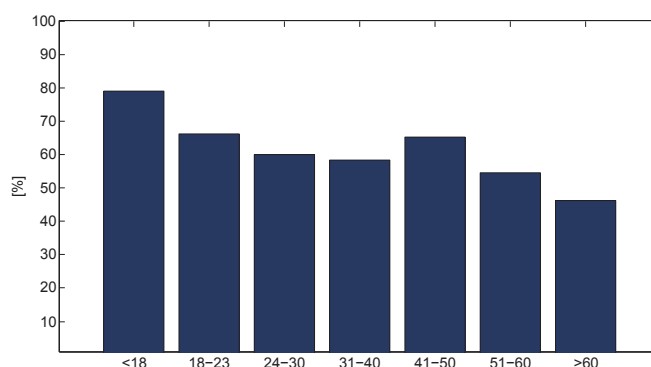


Figure 2.9.: Percentage of participants that either already use a PIN authentication or would use biometrics if available (for each age group).

2.5. Summary

This chapter provided an overview of the existing and currently developed authentication methods for mobile devices. Most phones do still only offer authentication via PIN. Mainly users of Android phones do have alternatives. From the beginning, there has been the option to draw a visual pattern for authentication, and with the newest Android version even face recognition is included. The presented authentication approaches show that there is continuous research activity, especially in the area of biometrics for mobile devices.

The survey results provided in Section 2.4 underline the need for research on user-friendly authentication methods for mobile devices. The main reason for people not to use the PIN method is the speed. Hence, the newly developed methods have to be very fast or, even better, run in the background and go unnoticed by the user. This requirement is met by accelerometer-based biometric gait recognition which is introduced in the following chapter.

CHAPTER 3

STATE OF THE ART OF BIOMETRIC GAIT RECOGNITION

Biometric gait recognition is the process of authenticating people based on the way they walk. Murray [103] was probably the first who detected subject-specific behaviours of walking individuals. In 1964 he stated “Although the excursions of both pelvic and thoracic rotation in repeated trials of the same subject were similar, there were striking differences in these excursions among the individual subjects tested”. And in a following publication he stated “if all gait movements are considered, gait is unique” [102].

Gafurov [48] divided research on biometric gait recognition into three groups based on the sensor which is used to record the gait:

1. Machine-vision-based
2. Floor-sensor-based
3. Wearable-sensor-based

The common advantage of all three approaches is that they are unobtrusive. The authentication process can be performed without any external intervention, making them very user-friendly methods. A short overview of the three approaches and their applications is given in the following sections. Although the machine-vision- and floor-sensor-based approaches are not relevant for smartphones, they indicate the uniqueness of this biometric characteristic and thus show the potential of gait for biometric authentication. A few additional approaches exist, analyzing acoustic signals from walking subjects (e.g. [130]).

3.1. Gait

Gait is “a manner of walking or moving on foot” [8]. The human walking pattern consists of multiple repeated gait cycles. Each gait cycle contains two steps. Figure 3.1 gives a schematic representation of one gait cycle and the measured vertical accelerations. During a stance phase the foot is on the ground, during a swing phase it is lifted and moved forward. Starting with a double support phase, where both feet are on the ground, the right foot is lifted and moved forward. This

right swing phase is followed by a second double support phase after the right foot touches ground again. Afterwards the left foot is lifted from the floor (left swing phase). The gait cycle ends, when the left foot touches the ground again.

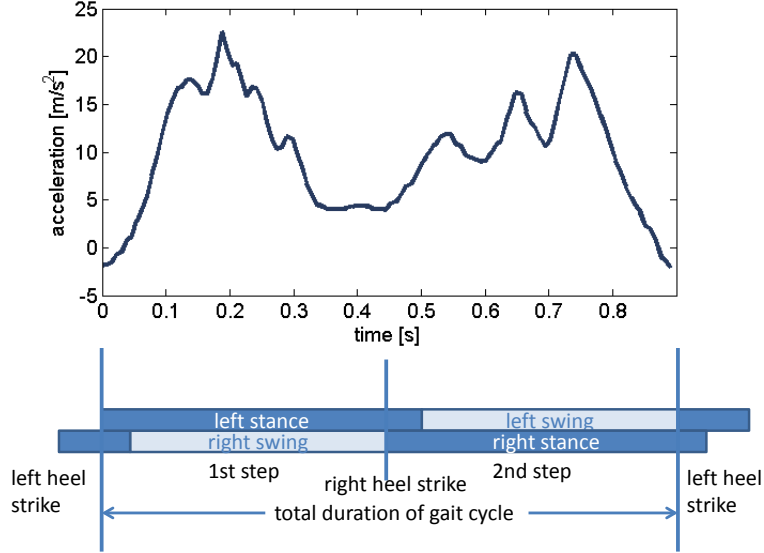


Figure 3.1.: Phases of a gait cycle (based on [103]) and the corresponding measured accelerometer values in vertical direction when the sensor is attached to the right hip of the subject (see Section 7.1).

Figures 3.2 and 3.3 show the vertical acceleration (including gravity) collected of two different walking subjects while the phone is attached to the right side of the hip. The collected signals illustrate the potential of gait recognition for biometric authentication. Regarding the data of one subject, a cyclic repetition is visible. Comparing the data of both subjects shows that this cyclic pattern has a subject-specific characteristic.

3.2. Machine-Vision-based Biometric Gait Recognition

In machine-vision-based biometric gait recognition subjects are recorded using video cameras. Because of the large variety of different applications for vision-based gait analysis, research in this area is well advanced. Applications are e.g. diagnosis of diseases, athletic performance analysis, surveillance, man-machine interfaces, content-based image storage and retrieval or video conferencing [4]. Although these pursue different goals, they have several processing steps in common, like background segmentation, detection of body joints etc. Some techniques require special markers attached to different body parts like knees or elbows, but in this section the focus will lie on marker-less techniques.

The significance of vision-based gait recognition is confirmed by several real-live applications. In [80], the Institute of Forensic Medicine in Denmark presents a checklist they use to find gait similarities and demonstrate it on the real case of a bank robber. In 2008 the BBC reported

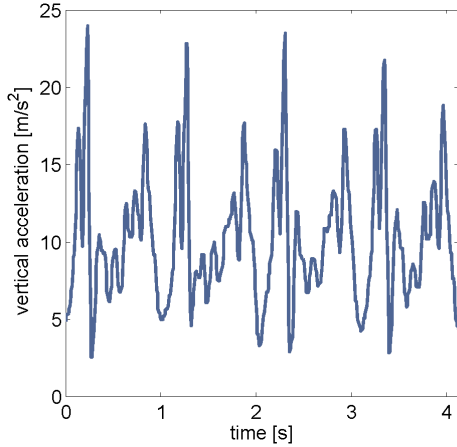


Figure 3.2.: Vertical acceleration recorded when subject A is walking.

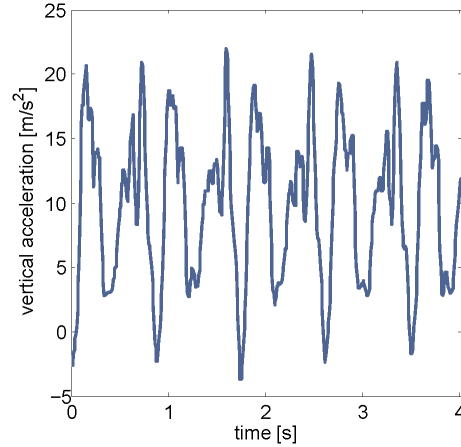


Figure 3.3.: Vertical acceleration recorded when subject B is walking.

that a burglar has been jailed over unusual walk [12]. Although in this case the identification has been conducted by an expert podiatrist, this shows a successful application of vision-based biometric gait recognition in surveillance. In contrast to most other application scenarios of machine-vision-based biometric gait recognition where the subject is interested in being monitored, the surveillance scenario has an additional challenge. Occlusions, e.g. by other subjects or constructional constraints, may prevent a complete capturing of the subject and impede the authentication.

Boulgouris et al. [20] give a comprehensive discussion about the state of the art in vision-based gait recognition up to 2005. They classify the approaches for feature extraction into model-based and holistic. In holistic approaches no specific underlying model is assumed and the features mainly rely on the extracted silhouette. In [77] the width of the silhouette is used. In [21] the silhouette is divided into angular sectors, and features are based on the mean distances of the foreground pixels to the center of the silhouette. Model-based approaches assume that the human walk is based on an underlying model. In general, these methods are more complex but they have the advantage of being more robust to noise, changes in viewing angle and changes in clothing. The extracted features are e.g. static parameters (like height or distance between feet) [76] or ellipse parameters (where the ellipses correspond to body regions like head or lower leg) [82]. Although these features allow for a compact representation of the specific gait, they are prone to acquisition errors which prevent the correct extraction of the necessary parameters.

In 2001, the Human Identification at a Distance (HumanID) program was started and the HumanID Gait Challenge Problem, consisting of a large data set collected from 122 subjects, several experiments, and a baseline algorithm, was published [145]. Major research in the area of vision-based gait recognition has been done in the group of Carter and Nixon [114] at University of Southampton. They published a large database containing data of 118 subjects in 2004 [146]. Recently, they established the Multi-Biometric Tunnel, which allows to unobtrusively acquire various non-contact biometrics in a constrained environment, as existent e.g. in airports. It is used for collection of gait, ear and facial data using cameras at different positions while the subjects are

walking through the tunnel.

Recently, the focus in research moved to addressing new challenges like 3D videos, low-quality CCTV data or the influence of covariates like clothes or time, because many sophisticated algorithms exist, which show very good recognition performance on public available databases. Recent publications in these areas are presented in the following subsections.

3.2.1. 3D Videos

During the last years, research started on gait recognition using 3D video data. Ariyanto and Nixon [9] presented an approach modeling the lower body by using 3D point cylinders for the thighs and shins. The center of hip is the origin of the global coordinate system. Based on the angles extracted from the cylinder orientations gait kinematics are extracted. Frequency components of the kinematics compose the dynamic features. In addition, three structural features are computed: height, stride length and footprint pose. The latter is a novel gait feature, inherently unique to 3D vision-based gait recognition. The recognition performance was evaluated on a database collected in the Multi-Biometric Tunnel, containing four data sets for each of the 46 subjects. The best classification rate of 79.4%¹ was obtained using a combination of the center of hip and the kinematic features and applying a 1-nearest neighbor algorithm as classifier.

3D data captured using a high-resolution projector camera system is analyzed in [159]. A 3D model of the human body is fit to the captured data for four different posture conditions and the postures in-between are interpolated. Dynamic features (joint angles) as well as static features (lengths of body parts) are extracted and compared using the L2-distance. Evaluation on a database including six subjects yielded a genuine accept rate of 100% when combining the dynamic and static features.

3.2.2. Surveillance

The suitability of using gait recognition in surveillance is evaluated in [19]. Using a frame differencing approach a motion map is created. To localize hips, knees, and ankles a Haar-based template matching method is applied and the results are amended using gait kinematic and anthropometric knowledge. The coordinates of the joints are computed in the coordinate system having the left ankle as origin (frontal video images are considered) and are normalized by the subject's height. These coordinates form the feature vector. Two different videos are compared via *Instantaneous Posture Matching*: The Euclidean distances between feature vectors from different frames in a specific time window are computed. The authentication is based on comparing the minimum distance with a threshold. No test results for the available CCTV database are given. Using CASIA-B² database taken under laboratory conditions an EER of 11.6% was obtained.

Akai et al. [7] focus on the robust authentication when only low frame-rate videos are available like in CCTV. They apply a super-resolution technique to generate a high frame-rate image sequence from the low frame-rate input. The problematic estimation of the initial phase is improved by applying continuous Dynamic Programming between the low frame-rate input video and a high frame-rate sample video. To minimize alpha-blending effects, a morphing-based interpolation between pairs of adjacent images is applied.

The main focus of Goffredo et al. [63] was the development of a self-calibrating view-invariant gait recognition system. Hip, knees, and ankles are identified from the silhouette. The angles

¹Unfortunately it is not clear from the publication to which standardized error rate this value corresponds to.

²<http://www.cbsr.ia.ac.cn/english/Gait%20Databases.asp>

between the thighs and the vertical axis as well as between the shins and the vertical axis are computed and projected onto the lateral plane. This projection is constructed in such a way that the viewpoint is rectified. The algorithm is tested using different databases and the results are compared to state-of-the-art methods. It is shown that, although the results for side-view walking are worse, the proposed algorithm achieves a Correct Classification Rate (CCR) of more than 50% when reference and probe are recorded from different viewing angles. Hence, it is outperforming the compared algorithms for which the CCR greatly drops when the angles differ by more than 20 degrees.

With the progress of face-recognition based on low-resolution images, it is possible to fuse gait and face recognition from surveillance videos. Evaluations with humans have shown that this kind of fusion advantages the face- or body-only recognition performance [119].

3.2.3. Influence of Covariates

In [88] the influence of time to vision-based gait recognition is evaluated. The other covariates (clothes, environment) are kept stable for the data collection which took place in four sessions during nine months. 21 subjects participated in all of the sessions, wearing overalls and no shoes during each session. To analyze the influence of clothing, on the last session part of the data was collected while the subjects were wearing their normal clothes. Combining data from side, front and top-view, the recognition performance dropped only by 5% over nine months. On the other hand, the influence of clothing was clearly visible. Comparing same-day data where subjects were wearing the same clothes a recognition performance of 100% was obtained. Comparing the probe data extracted when the subjects were wearing normal clothes with the same-day references obtained from subjects wearing overalls, decreased the recognition performance to around 40%.

Boulgouris et al. [20] implemented a silhouette feature and applied different comparison algorithms like Dynamic Time Warping (DTW) or HMMs. Evaluation was performed using the database collected by University of South Florida (USF)³, which contained samples collected on different surfaces (grass and concrete). For all comparison algorithms a large decrease of the Cumulative Match Score (CMS) was reported. When probe and reference were recorded under different viewing points (left vs. right) a CMS of 90% was reported which dropped to 18% when viewing angle and surface changed.

3.3. Floor-Sensor-based Biometric Gait Recognition

The main applications for floor-sensor-based biometric gait recognition are access control and smart homes. In addition it is used for clinical gait analysis, like in the BTS Gaitlab [26] in combination with video-recordings. Sensors can be integrated directly into the floor or installed in mats. A good overview is given in the Encyclopedia of Biometrics [83], where the technology is listed under the keyword *Footstep Recognition*.

The integrated sensors are either pressure sensors, mainly measuring ground reaction forces, or binary sensors which only report if they notice a footstep or not. The first work in this area was reported in 1997 by Addlesee et al. [3]. They assembled the ORL Active floor consisting of a squared arrangement of nine tiles sensing vertical ground reaction forces with a sensitivity of 50 grams. A database was created containing twenty footstep traces of 15 different subjects. HMMs

³<http://figment.csee.usf.edu/GaitBaseline/>

were used for classification which were trained with one half of the traces and tested using the other half. Best settings resulted in correct classification of 137 out of 150 traces (91.3%).

In the following, similar floor systems like the Smart floor [118], Ubi-floor [163], EMFi Floor [139] or InfoFloor [138] were built. Also the group around Nixon and Carter at Southampton University started to work on floor sensors in addition to their work on vision-based gait recognition [91].

In [29] features from force sensors and from videos are combined and an EER of 1.6% was reported. Vertical ground reaction force is recorded while subjects were walking on three wooden tiles which had piezo sensors in each corner. At the same time they were filmed from the side. Power Spectral Density was computed from the ground reaction force and three different features were extracted from the subject's silhouette (i.e. mean, variance and Fast Fourier Transform (FFT) of the histogram). In addition, a temporal template was created by summing up all silhouette images of one gait cycle. The test database contained samples of 16 subjects. Although the EER of the histogram mean was slightly lower than that of the fused features, they identified the latter ones as the best choice due to the higher discriminatory power which was measured as the area between the horizontal line through $EER + 5\%$ and the False Accept Rate (FAR) and False Reject Rate (FRR) curves.

The most recent presented system is Ubi-floor II [162] consisting of 1536 photo interrupter sensors regularly distributed over 24 wooden tiles. Two different classes of features are defined. Walking patterns describe spatio-temporal variations which are extracted from a sequence of an individual's footsteps, like stride length or foot angle. Stepping patterns are defined as temporal variations extracted from one single footstep, e.g. transitional footprint. The features were transformed to vectors containing a "-1" if the corresponding sensor had not been covered by the foot and a "1" if the sensor was covered. Data of ten subjects was collected; each of them provided 50 walking samples. Of each walking sample five footsteps were used to compute the features. A recognition accuracy of 99% was obtained when training Multilayer Perceptron Networks separately for walking patterns and stepping patterns and performing fusion on score level.

3.4. Wearable-Sensor-based Biometric Gait Recognition

The wearable-sensor-based approach is the most recent one. An overview of all relevant publications using accelerometers for data collection is given in Table 3.1, together with the reported EER, CCR or Genuine Match Rate (GMR). It is indicated in column *Setting* whether enrolment and probe data are collected on the same day, on different days (cross-day) or if the database consists of data of two different sessions but enrolment and probe data are taken at least partly from the same session (mixed-day). This is an important factor to consider, as there is a high variation of gait over time, resulting in much lower error rates when enrolment and probe data are from the same day compared to the cross-day scenario.

In the beginning, wearable sensors like gyro sensors or accelerometers have been used for medical gait analysis. In 2004, Morris [97] used a GaitShoe system consisting of several sensors (bend sensors, ultrasound, accelerometers etc.) attached to a shoe. The main goal was to extract clinical relevant information from the sensor. But in addition to distinguish between subjects with Parkinson's disease and without, he also analyzed if subjects can be recognized based on their gait. 25 features were extracted from the different sensor outputs and a correct classification rate of 97.4% was obtained using Neural Networks and data of ten subjects.

Initial research on gait recognition using only accelerometers was concentrated on two research groups: VTT Technical Research Centre of Finland and Norwegian Information Security laboratory

(NISlab) at Gjøvik University College. Their work is presented in the following two subsections. Subsection 3.4.3 gives an overview over further publications in this area.

3.4.1. VTT Technical Research Centre of Finland

The first researchers which really focused on subject authentication using accelerometers were from VTT Technical Research Centre of Finland [6]. Their targeted application scenario was the authentication on mobile devices but they used a dedicated accelerometer for data collection. The sensor was attached to the waist of the subjects at the back. In addition the subjects had to carry a laptop which was used to record the data. The signals were recorded with 256 Hz frequency. Accelerations in three directions were measured but only forward-backward and vertical acceleration were used in the evaluation. From these signals single steps were extracted by searching local minima and maxima. First the average length⁴ of one step is computed. Minima and maxima are searched alternately within the one step length interval starting at the previous maximum or minimum. As the minima and maxima are ambiguous, mean points are used as end and starting points of steps. These are obtained by using a data block which has length three times the step length and subtracting the average of this data block from the data contained in this block. The mean points are then the transitions from positive to negative values after a maximum. The block is moved, the average step length is updated and the next mean points are computed. The resulting steps are normalized in length. Cross-correlation is used to determine the most similar steps which are averaged and normalized in amplitude. These averaged steps from acceleration in forward-backward and vertical direction form the gait code. Distances between two gait codes are computed using cross-correlation. Test data was recorded from 36 subjects at two different days. In the first session, each subject walked 20 meters. In the second session the same distance was covered three-times. An EER of 6.4% was obtained.

In [94] an extended database is used, including walking in normal, fast, and slow velocity. In each session each subject walked 20 meters once in each velocity. In the evaluation different features are opposed to the previously applied method using cycles and cross-correlation. The FFT is calculated for vertical and forward-backward acceleration with a sample frame of 256 and 128 samples frame overlap. The feature vector consists of the first 40 FFT coefficients per channel. For the histogram method, 10-bin histograms were normalized by the length of the data measured in vertical- and forward-backward-direction and concatenated to give the feature vector. The last method computed third and fourth order moments for the two acceleration directions, their concatenation gives the feature vector. The distance methods applied are not described in the publication. The new features did not further improve the recognition rates stated in [6]. In further publications VTT proposed to fuse voice and gait on a score level [148] or even further fuse these results with fingerprint in case a high security level is required [147]. While these specific multi-modal approaches are theoretically of interest they do not provide an acceptable usability.

3.4.2. Norwegian Information Security laboratory at Gjøvik University College

In 2005, the Norwegian Information Security laboratory (NISlab) started their research on accelerometer-based biometric gait recognition [131]. Gafurov wrote several publications in this area which are contained in his PhD-thesis titled “Performance and Security Analysis of

⁴The terms step or cycle length refer to the duration in the whole thesis, not the covered distance.

Publication	Sensor	Sensor Position	# Subjects	Setting	Best EER
Ailisto [6], 2005	dedicated	waist at back	36	c	6.4%
Mäntijärvi [94], 2005	dedicated	waist at back	36	c	7%
Vildjiounaite [148], 2006	dedicated	three different	31	c	13.7%
Gafurov [55], 2006	dedicated	ankle	21	s	5%
Gafurov [61], 2006	dedicated	hip	22	s	16%
Gafurov [54], 2006	dedicated	ankle	21	s	5%
Rong [125], 2007	dedicated	waist at back	21	c	5.6%
Gafurov [58], 2007	dedicated	trouser pocket	50	s	7.3%
Rong [126], 2007	dedicated	waist at back	35	c	6.7%
Vildjiounaite [147], 2007	dedicated	hip and breast pocket	32	c	13.7%
Gafurov [49], 2007	dedicated	hip	100	s	13%
Gafurov [59], 2007	dedicated	hip	100	s	13%
Holien [70], 2007	dedicated	hip	25	s	18%
Stang [133], 2007	dedicated	waist	13	s	26%
Holien [69], 2008	dedicated	waist	60	m	5.9%
Gafurov [56], 2008	dedicated	ankle	30	s	5.6%
Gafurov [62], 2008	dedicated	arm	30	s	10%
Gafurov [57], 2009	dedicated	four different	30	s	5%
Pan [120], 2009	Wii remote	waist	30	c	80.1% GMR
Derawi [39], 2009	dedicated	waist	60	m	9.19%
Sprager [132], 2009	phone	waist	6	c	92.9% CCR
Mjaaland [93], 2009	dedicated	hip	50	s	6.2%
Bächlin [13], 2009	dedicated	ankle	5	m	21.3%
Trivino [142], 2010	unknown	unknown	11	s	3%
Gafurov [53], 2010	dedicated	ankle	10	c	59% GMR
Gafurov [60], 2010	dedicated	ankle	30	s	1.6%
Bours [22], 2010	dedicated	hip	60	m	1.6%
Wang [152], 2010	dedicated	waist at back	24	s	5%
Frank [47], 2010	phone	trouser pocket	25	s	100% CCR
Kwapisz [79], 2010	phone	trouser pocket	5	s	100% CCR
Derawi [40], 2010	dedicated	waist	60	m	5.7%
Derawi [41], 2010	phone	waist	48	m	20.1%
Yan [160], 2010	dedicated	waist at front	10	s	6.29%
Gafurov [51], 2010	dedicated	hip	100	s	7.5%
Mjaaland [92], 2011	dedicated	hip	50	s	6.2%

Table 3.1.: Overview of publications using only accelerometer data for gait recognition. The best EER obtained for the basic settings (no spoofing, concrete floor etc.) is reported. Setting: s=same-day, c=cross-day, m=mixed-day; CCR and GMR indicates that the stated performance measure is the correct classification rate or the genuine match rate instead of EER.

Gait-based User Authentication” [50]. He used a dedicated sensor which was attached to the ankle [54], hip [59], arm [62] or placed in the trouser pocket [58] of the subjects (see Figure 3.4). Equal error rates for the four different sensor locations are given in Table 3.2.



Figure 3.4.: The sensor used by Gafurov attached to different body parts of the subject, taken from [57].

Sensor Placement	EER	# Subjects
Ankle [54]	5%	21
Hip [59]	13%	100
Trouser pocket [58]	7.3%	50
Arm [62]	10%	30

Table 3.2.: Results obtained by Gafurov for different sensor placements.

Gafurov also focused on the robustness of accelerometer-based gait recognition against impersonation attacks. In [49] he showed that the FAR is increased if impostor comparisons are only performed on data of subjects having the same gender. Gait mimicking was considered in [59]. 90 subjects were randomly paired and were told to study and imitate their partner’s gait. Two different scenarios were considered. In the first one, the mimicking subject was walking directly behind the one to imitate. In the second scenario this visual help was not available. The FRR obtained when comparing the normal walk of a subject to the one when he is mimicking somebody else’s walk was significantly higher than the previously computed, where two regular walks of the subject were compared (friendly scenario). This shows that the subjects were able to change the walking style; hence, it is possible to circumvent the authentication. But although they changed their walk, the overall EER was not significantly decreased, showing that it is hardly possible to successfully mimic somebody else’s gait. A further evaluation showed that the EER is greatly increased if the impostor probe is only compared to the closest reference in the database (closest person attack).

The ability to mimic gait is also analyzed by Mjaaland in [92] and [93]. The applied method yielded an EER of 6.2% on a same-day database with 50 subjects. Six subjects took part in the short-term hostile scenario. In five different sessions during two weeks they tried to mimic the gait of one single victim. During each session they obtained feedback via statistics and videos. One additional subject took part in the long-term hostile scenario, following the same process but over a time period of six weeks. In general, the subjects were not able to learn the victim’s gait.

In [53] Gafurov showed that there is a large difference between same-day comparisons (genuine recognition rate of 80-90%) and cross-day comparisons (genuine recognition rate of 26-59%). The impact of different shoes is analyzed in [56]. Data of 30 subjects were collected which were wearing the same four types of shoes (see Figure 3.5). Analysis was performed for each acceleration direction separately and it was shown that, in case the sensor is attached to the ankle, side-way acceleration performs best. Shoe type B resulted in the lowest EER of 5.6% and shoe type C performed worse (EER = 15.0%). When data of different shoes were compared the EER increased significantly to up to 32.9%, obtained when data of shoe A is compared to data of shoe D.



Figure 3.5.: Different shoe types used by Gafurov, taken from [56].

The evaluation of [56] was performed again with an improved comparison algorithm [60]. The lowest EER of 1.6% was obtained with shoe type A, which previously had an EER of 7.2%. By applying this method to the database used in [59] the EER could be decreased from 13% to 7.5% [51]. In [52] the results of [60] could be further improved to an EER of 0.5% by fusing the scores obtained for the different acceleration directions.

The work in [69] by Holien focused on developing a robust cycle extraction method by considering several parameters (e. g. lengths between peaks, number of local maximum points) when detecting the cycle starts. On a database with 60 subjects, the best reported mixed-day EER was 5.9%. This was obtained by normalizing the cycle lengths to 100 samples per second, adjusting the acceleration values depending on the normalization factor, and using the median cycles as reference and probe. DTW is used as distance metric. Derawi et al. [40] apply the same cycle extraction method, but instead of using the median cycle all cycles (except outliers) were kept. These are compared using a cyclic rotation metric, slightly decreasing the EER to 5.7% (on the same test database).

Bours and Shrestha [22] enhanced three of the existing cycle extraction methods presented in [59], [69], and [40] by applying Principal Component Analysis (PCA) to the average cycles and only keeping those eigenvectors which contain 95% of the variance of the data. This decreased the EER to around 1.6% for all evaluated combinations of cycle extraction and comparison algorithm.

3.4.3. Further Research

In addition to these two main research groups, several other researchers published articles in the area of accelerometer-based biometric gait recognition, but for most of these it does not seem to be the main focus of their work, and the evaluations are mainly performed on small databases containing only a few subjects.

Rong [125, 126] applied a very simple cycle extraction method based on the number of zero-crossings. Their best results were obtained by normalizing the detected cycles to equal length by applying a DTW algorithm and computing the average cycle. The gait template consists of the

concatenated average cycles for each acceleration direction. The same DTW algorithm is used for comparison, and an EER of 5.6% was obtained on a database of 21 subjects [125].

In [120] accelerometers (Wii remote) were attached to five different parts of the body, i.e. upper arm, wrist, thigh, ankle, and waist. The single channels resulted in recognition rates between 66.8% for the wrist and 74.5% for the upper arm which could be increased to 96.7% when combining all five channels. The reported results are cross-day-results obtained on a database containing 30 subjects.

The first work stating results for a database collected by using a mobile phone (Nokia N95) is by Sprager and Zazula [132]. The phone was attached to the hip and the recorded forward-backward and vertical acceleration data was divided into cycles. Cumulants of order 1 to 4 were extracted from the signals and transformed into a feature vector. SVMs were used for classification. Using a cross-day test database of six subjects, they obtained a recognition rate of 92.9%.

Bächlin et al. [13] evaluated the influence of shoes, weight, and time on gait. Gait of five subjects was recorded while they were walking on a treadmill to have controlled walking speeds. Four different feature types are computed for different signal types: segments containing 64 data samples with and without normalization of step length, and FFT coefficients from a jumping window containing 256 samples starting at regular points or at the heel strike. Similarity between two data sets is computed via one-way Analysis of Variance (ANOVA) and by determining the percentage of positions in the feature vector for which the ANOVA showed that they are not statistically significant different. When reference and probe data were obtained from the same task (speed/weight) and on the same day, an EER of 2.8% was reported. It increased to 21.3% when mixed data of several days was used and even increased to 31.5% for data of several tasks (same-day).

Trivino et al. [142] propose a different approach using a linguistic model based on the computational theory of perceptions. The perception of the signal evolution in one gait cycle was modeled by a Fuzzy Finite State Machine (FFSM) and the model is expressed via linguistic terms. The method was tested on a database containing same-day data of eleven subjects and an EER of 3% was obtained.

In [152] Wang et al. apply a maximum-based cycle extraction to the vertical acceleration measured when the sensor is attached to the back of the waist. The cycles are segmented based on identified extreme points, and different features, like relative time in cycle or slope of straight line between two endpoints, are extracted. Classification is based on the DTW distance. Evaluating the approach on a same-day database containing data of 24 subjects resulted in an EER of 5%.

Frank [47] also uses acceleration data collected via a smart phone placed in the trouser pocket to identify subjects. Between 12 and 120 seconds of data were collected from 25 individuals. For each subject time-delay embedding models were created, and the probe segments were mapped into the model-space. Considering four nearest neighbors, scores are calculated for each mapped test segment. Classification is based on the highest average score, resulting in a perfect classification result for the given test set. On his web page [46], Frank has published his code and a further data collection he made. This code was evaluated on database 2 in Section 7.6 and benchmarked to the algorithms contributed in this thesis.

The focus of the Wireless Sensor Data Mining (WISDM) Project at Fordham University is mining data collected from various sensors (accelerometers, Global Positioning System (GPS) sensors, cameras etc.) of smartphones or other mobile devices. The project group is led by G. Weiss and consists of over a dozen members, mainly undergraduate students [45]. In 2010 they published a method for user identification on smartphones using decision trees and neural networks [79]. Data was collected from 36 subjects who carried a smartphone in the front trouser pocket.

The smartphone was used to record accelerations while the subjects were walking, jogging, walking up-, and downstairs. 43 features, like average resultant acceleration, binned distribution and time between peaks, were extracted from ten seconds segments extracted from the acceleration data to form the feature vector. For the authentication task one model was generated for each subject. Using all available testing data (collected on the same day as the training data) a 100% GMR was reported. It was obtained by using the *most frequent user strategy*, where each segment is classified separately and the most frequent predicted subject is assigned to the whole data set. Verification results were only reported for five subjects. The False Non Match Rate (FNMR) lies between 7.1% and 17.9%, and the False Match Rate (FMR) lies between 2.9% and 7.4% for different subjects.

3.5. Summary

This chapter provided an overview of the state of the art in biometric gait recognition, separately for the three main groups vision-based, floor-sensor-based, and wearable-sensor-based. Approaches belonging to the first two groups are not relevant for authentication on smartphones. Nevertheless, they give an insight on the amount and quality of the methods and demonstrate the discriminative properties of human walking.

Because accelerometer-based biometric gait recognition is still a recent technology, a short presentation of all relevant literature was given. Several drawbacks exist in most of these publications which make further research necessary:

- In most of the publications, dedicated high-quality sensors are used for data acquisition, and the evaluation is performed on powerful desktop PCs. It is unclear how or if at all the presented methods perform on smartphones.
- Sensors located at the arm, ankle, back or front waist do not correspond to realistic positions of smartphones. Those are in general located in a pouch at the belt, in trouser pockets or bags (see Section 2.4).
- The sizes of the prototype sensors are far from being acceptable and do not allow for a comfortable usage (see Figure 3.4).
- In a real application, users would enrol once and then use the created reference data over a longer period, probably of at least one month. Hence, in general reference and probe data are not collected on the same day. In [53] it is shown that there is a high decrease in recognition performance when data of different days is compared instead of data from the same day. Nevertheless, in most publications same-day databases are used (indicated by the corresponding label 's' in Table 3.1).

The above analysis stimulated this research and will be considered to proceed towards the goal to enable gait authentication on mobile devices. The following chapter presents the design of an accelerometer-based gait recognition system for smartphones.

CHAPTER 4

DESIGN OF AN ACCELEROMETER-BASED BIOMETRIC GAIT RECOGNITION SYSTEM

The previous two chapters described different methods for authentication on mobile devices and for gait recognition in general. This chapter is presenting a more abstract view on the problem of designing an accelerometer-based biometric gait recognition system for smartphones. Section 4.1 presents the design of the Modular Biometric Authentication Service System (MBASSy). MBASSy was developed as functional prototype at the Center for Advanced Security Research Darmstadt (CASED)¹. Afterwards, the conceptual representation of a general biometric system as defined in ISO/IEC SC37 Standing Document 11 [73] is given, and the realization of the different components in MBASSy is indicated. In Section 4.3 the operating sequence of enrolment and authentication in MBASSy is described.

The current design of MBASSy meets the requirements of a system controlling the access to the smartphone. As a supplement to the PIN, it gives the possibility to choose between different authentication modules based on different biometric modalities. Thus, it provides significant benefit to the user as entering a PIN is limited to those events, in which the enrolled subject can not be recognized based on his biometric characteristics. The recent trend to integrate Near Field Communication (NFC) components into smartphones offers further application scenarios. For example, the authentication result generated by one of the MBASSy modules can be transferred to the NFC reader and be used to open doors as the enrolled user approaches them or to pay cash-less. In a similar manner, it might be used for authentication at external services.

4.1. Modular Biometric Authentication Service System

To facilitate the deployment and evaluation of different (biometric) authentication methods on mobile devices MBASSy was developed [157]. MBASSy is an Android application which provides an authentication framework. Different authentication methods can be implemented as separate modules and easily integrated into the system. Depending on the mode, either one or several modules have to return a positive authentication result to authenticate the user. The main

¹With kind support of the LOEWE research program.

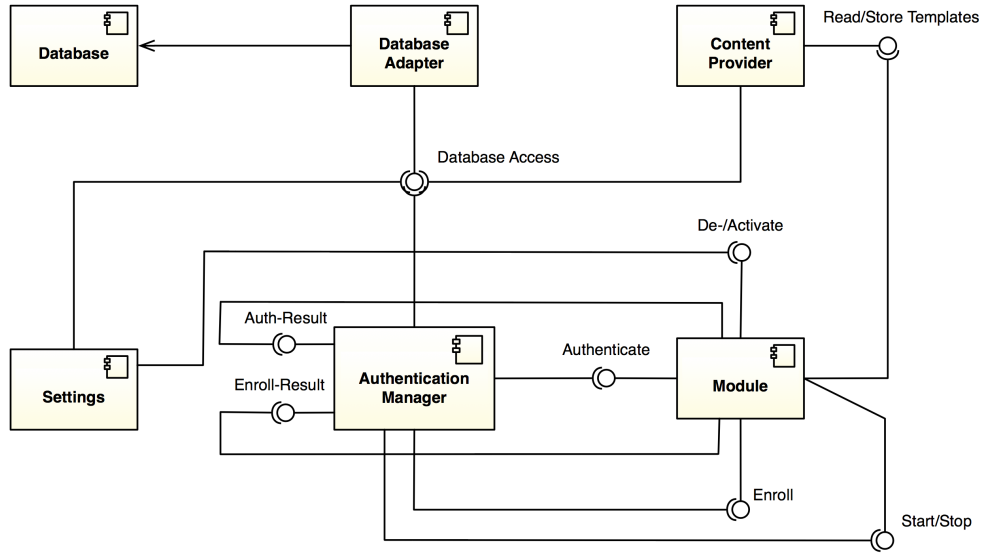


Figure 4.1.: MBASSy system overview.

components of MBASSy are explained in the following sections. Figure 4.1 gives a system overview.

An important tool for security in MBASSy are so-called tokens. They restrict database access and communication to the relevant modules. Module tokens are assigned to the modules on activation. Authentication tokens are generated during enrolment and authentication.

4.1.1. Database

The database is used to store information about the modules (e.g. activation status, priority level), the enrolled users (e.g. subject ID, time of enrolment), statistics (e.g. number of successful authentications), and links to the reference data which are stored in separate files due to space limitations. Instead of storing the reference data on the phone, it is also possible to store them on an external server. This has the advantage, that the same reference data can be used for different applications, e.g. for authentication on the smartphone and on the tablet PC. Storing data on an external server introduces several privacy and security risks. Because of the variability of biometric data, cryptographic encryption methods cannot be reasonably applied to prevent those. Therefore, template protection techniques, like [112], are necessary.

Access to the database is granted in two different ways. The application package may access the database without restrictions via a content provider, e.g. to update the activation status of the modules. Only a restricted access is available for the modules. While read access is possible at any time, modules are only allowed to write to the database when answering an enrolment request. Before database access is granted the correctness of module tokens and, in case of write access, authentication tokens is confirmed.

4.1.2. Modules

An MBASSy module is a distinct application package which contains all necessary code for a specific authentication method. Two different types of modules are distinguished: *foreground modules* and *background modules*. Background modules do not require any user interaction as they run in a background process. Foreground modules request the user to interact with the authentication process. Modules provide methods for enrolment and authentication (see below) and can be activated or deactivated. Background modules also offer methods to start and stop the background activities, like collection of accelerometer data. On activation, a token is assigned to the module which is later used in the communication between the system components, e.g. to assure that a module can not access the data of other modules.

4.1.3. User Interface

The user interface provides a unified access to the available modules (see Figure 4.2 for screenshots). In the category *module settings* all activated and further available modules are listed, divided by background and foreground modules. It is possible to change the activation mode and to access the module specific settings. The category *users* lists all enrolled users, offers the possibility to add further ones, or to enrol the users for different modules. When a new user is added, he has to choose a password as fallback authentication for the case no positive authentication is returned by the modules. *Preferences* offers different MBASSy settings like the authentication mode (verification or identification). The category *system info* provides information which is necessary for module testing and development. This is for example, the number of templates in the database and the list of available sensors. The possibility to export the database is also offered for an external evaluation of the recorded data.

4.1.4. Background Service

The background service is the basis of the system and responsible to handle enrolment and authentication requests. It implements a broadcast receiver to be able to react on the *screen on* event and request the authentication results from the activated modules.

4.2. Components of a General Biometric System

The main focus of MBASSy is biometric authentication. Therefore, it is necessary to pay attention to the ISO/IEC SC37 standard describing the general components which are comprised in a biometric system. Figure 4.3 displays these components and the workflow operated at enrolment, verification and identification. The following sections give a short description of the components and indicate how these are realized in MBASSy.

4.2.1. Data Capture Subsystem

The subject presents his biometric characteristic to the data capture subsystem. Depending on the biometric modality the sensor is e.g. a camera or a fingerprint reader. In case of MBASSy, the available sensors depend on the hardware of the smartphone. For biometric gait recognition accelerometers are necessary; face recognition requires a high-quality camera.

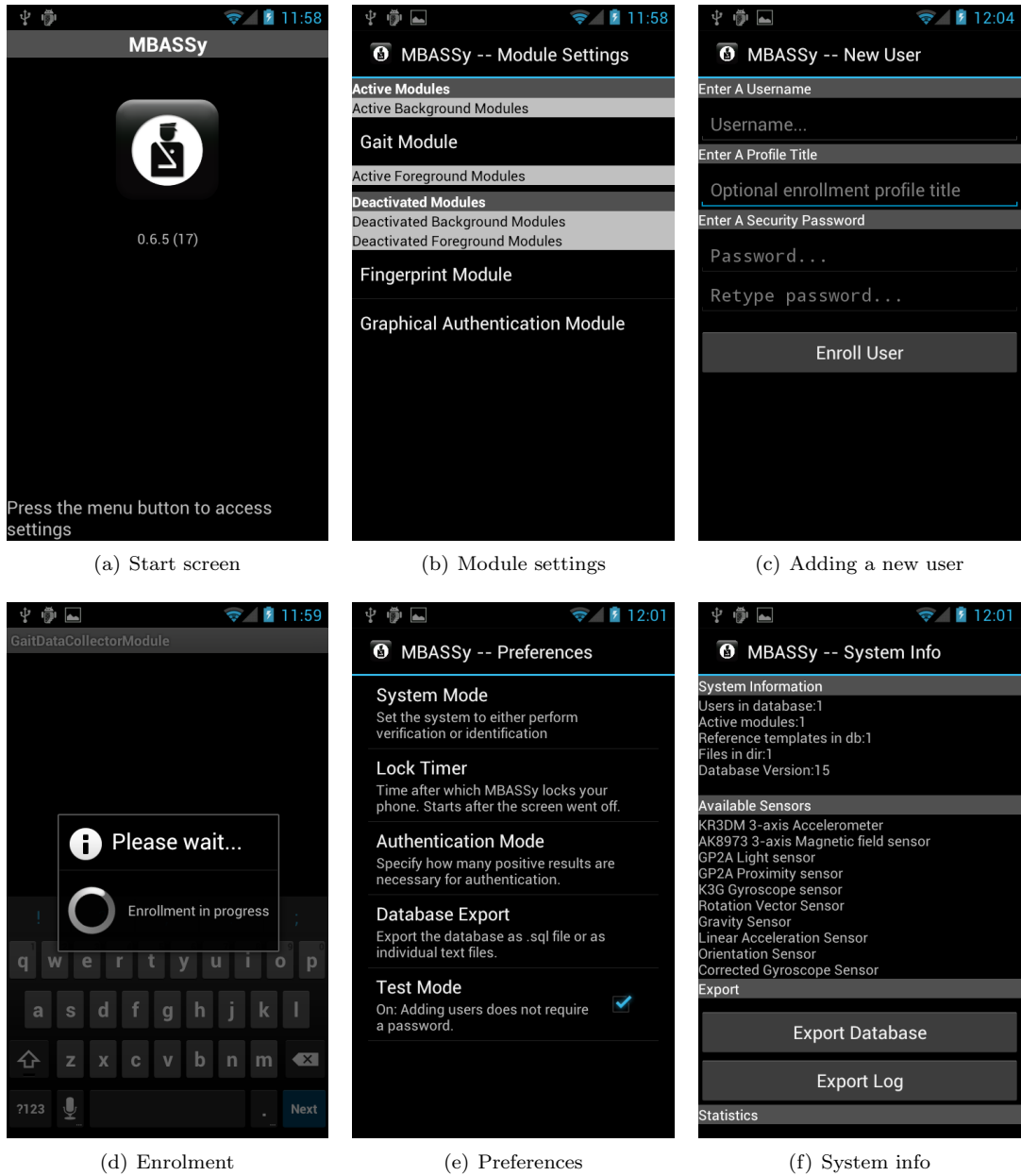


Figure 4.2.: MBASSy screenshots.

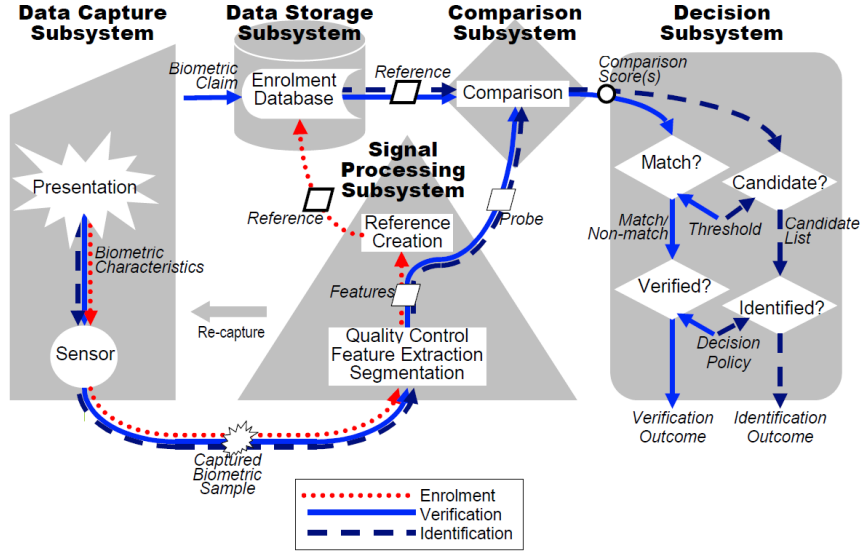


Figure 4.3.: Components of a general biometric system (as presented in ISO/IEC SC37 Standing Document 11 [73]).

4.2.2. Signal Processing Subsystem

The captured biometric sample is transferred to the signal processing subsystem. Features are extracted, and depending on the task of the system, either the reference or the probe is composed. The reference is created once during enrolment and then stored in the data storage subsystem. In case of authentication methods using machine learning algorithms (like HMMs), signal processing involves training of the model, and the stored reference might also be such a model. During each authentication, the signal processing subsystem creates the probe which is used in the comparison subsystem. When developing authentication methods for MBASSy, it is important to consider the low computational power of the smartphone, because all signal processing steps need to be performed on the phone.

4.2.3. Data Storage Subsystem

The data storage subsystem contains the enrolment database. It holds the references of all enrolled subjects and might also contain some additional information, like subject ID or date of creation. MBASSy contains a SQLite database as described above. Because in general only a small amount of subjects uses the identical phone, this database will only contain a low number of entries. It is possible to store multiple references per subject and application. For accelerometer-based gait recognition it might be necessary to store different references for different walking velocities.

4.2.4. Comparison Subsystem

In the comparison subsystem probe and reference data are compared, and a score is calculated. In verification mode only the reference corresponding to the stated ID is used (1:1 comparison).

In identification mode the probe is compared to all available references (1:n comparison). For machine learning authentication methods, e.g. SVMs, the result of this comparison might already be the classification result instead of a score.

4.2.5. Decision Subsystem

Based on the calculated comparison score(s) the authentication decision is generated. For most systems, this involves only the comparison of the scores with an application specific threshold. In verification mode, the subject's identity is verified if the (distance) score is below that threshold. In identification mode, the determined ID is the one belonging to the reference which is most similar to the probe, preconditioned that the corresponding (distance) score is below the threshold.

4.3. Functions of the Modular Biometric Authentication Service System

As indicated in Figure 4.3, each biometric system can run in three different function modes: enrolment, verification and identification, where the latter two can be grouped by the term *authentication*. The implementation of the different modes in MBASSy is described in this section.

4.3.1. Enrolment

On creation of a user account, the user can be enrolled for the active modules. He can either choose each module separately or choose "enrol for all". MBASSy sends an enrolment request to the specified modules. An enrolment Graphical User Interface (GUI) is implemented in each module, guiding the user through the necessary steps. For example, enrolment at the gait authentication module requires that the subject is walking for a specific time. The current module indicates start and end of the walking time by an acoustical and vibration signal. A face recognition module would request the user to take a picture of his face. Signal processing is implemented in the specific module to calculate the reference data from the captured sample. The enrolment is finished when the reference template is stored in the database.

4.3.2. Authentication

Because MBASSy is currently designed to control access to the smartphone, the authentication request is bound to the screensaver. The formal process is depicted in Figure 4.4. If the phone is not used, the screensaver will be activated. If the user deactivates the screensaver, MBASSy starts the authentication process. Starting from the highest ranked background module and ending at the lowest ranked foreground module, each module is asked for an authentication result. The process stops if enough results are collected (e.g. if only one of the modules is required to deliver a positive results, the process can stop after receiving the first positive result).

As stated before, background modules do not require any user interaction. A continuous authentication is implemented for the gait module. This means, that the module starts collecting the necessary accelerometer data after the screensaver has been switched on. When the needed amount of data is collected (e.g. 20 seconds) features are extracted and compared to the reference data. Based on the comparison score the authentication result is obtained and stored in the database. Afterwards, the data collection starts again and the process is repeated. When MBASSy

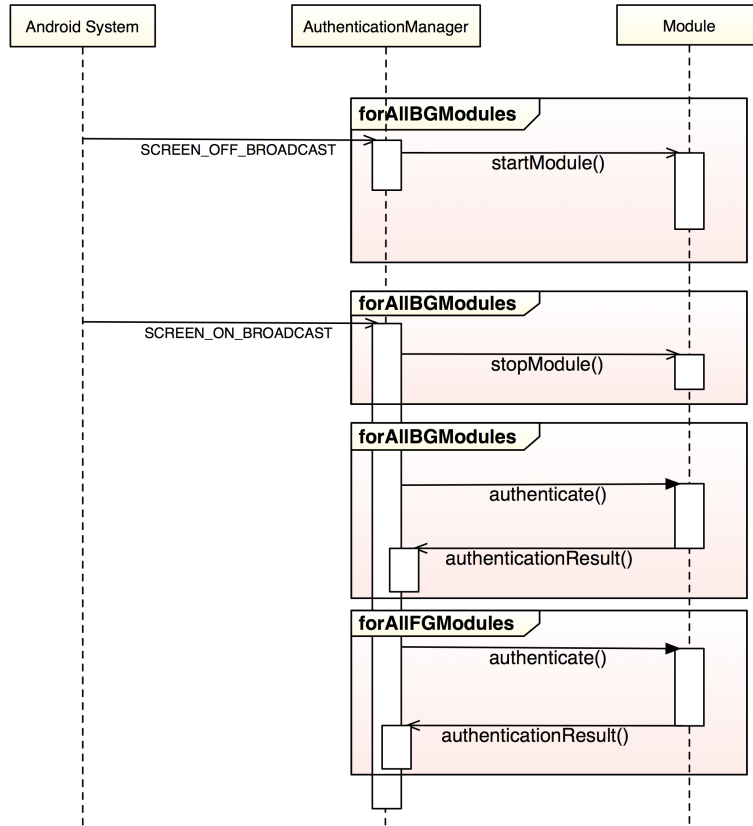


Figure 4.4.: The authentication process performed in MBASSy.

requests an authentication result from the gait module, the only necessary step is to get the last authentication result from the database and return it to MBASSy. This happens very fast, such that the user does not notice any delay in deactivating the screensaver. Hence, he is authenticated without any impairment compared to the no-authentication-case.

Foreground modules on the other hand require user interaction. When an authentication result is requested from a foreground module, its authentication GUI is displayed. Similar to the enrolment GUI it guides the user through the authentication process, e. g. by requesting a photo of the finger. The extracted probe feature vector is compared to the reference and the authentication result is returned to MBASSy.

The verification mode requires one user to be set as *active user*. Only reference data of the active user will be considered during authentication. In identification mode, the probe is compared to all stored references of the respective module. The authentication is positive, if the probe data matches one of the stored references.

4.4. Summary

This chapter presented the design of a biometric authentication system for smartphones called MBASSy. The components of a general biometric system according to ISO/IEC SC37 Standing Document 11 are described as well as their realization in MBASSy. The designed system offers a unified environment for an easy deployment of different authentication methods. Although designed for biometric authentication, modules offering e.g. graphical authentication can also be integrated. The user may choose his favorite methods and create a personalized authentication system for his needs. Currently, several modules exist, like continuous gait recognition, graphical authentication, finger photo authentication, and speaker recognition (as foreground and background module).

This chapter describes the gait recognition methods which are developed for this thesis and are subject to the evaluations in Chapter 6 to 8. In the context of vision- and floor-sensor-based gait recognition, Cattin stated that feature extraction methods can be divided into two groups: systems that need to locate one step or a complete gait cycle and systems that do not need to locate the gait cycle within the acquired data [29]. To classify the features, there are again two different approaches: template-based or stochastic classification [20].

For accelerometer-based gait recognition similar four categories exist (see Figure 5.1), where stochastic classification is replaced by the more general term *machine learning*. After preprocessing, the data are either segmented based on gait cycles or by using fixed-length time segments. Feature extraction is an optional step. The common two approaches are either to use the raw data extracted from gait cycles directly, or to apply a fixed-length time segmentation and extract features from these segments. Classification by machine learning algorithms is mostly used in combination with non-cycle-based features, whereas the gait cycles are compared using template matching algorithms. This combination is therefore referred to as *cycle-based method*.

The preprocessing, described in the following section, as well as the segmentation and feature extraction presented in Sections 5.2 and 5.3 are part of the signal processing subsystem presented in Section 4.2.2. The proposed classification algorithms belong to the comparison subsystem (see Section 4.2.4).

5.1. Preprocessing

Before the segmentation process can be started, the data have to be preprocessed by applying the following steps, see Figure 5.2.

Walk Extraction To process only meaningful data, it is necessary to identify those parts of the data which are collected while the subject is walking. In general, this can be done by applying activity recognition methods as in [25, 71]. The data used in this thesis are collected in a constrained environment; therefore, this process can be facilitated. Figure 5.3 gives an example of the captured data, taken from database 1 (see Section 6.1). For the data

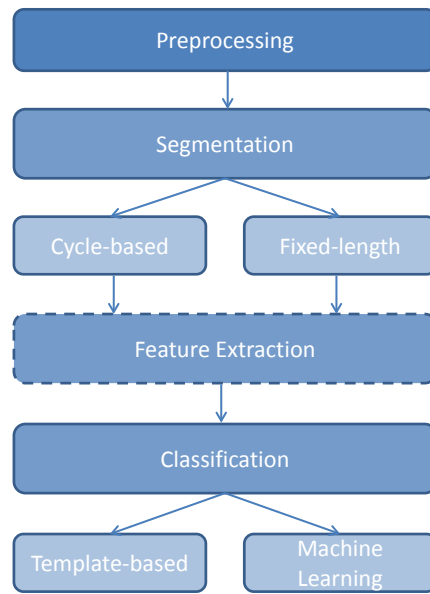


Figure 5.1.: Different approaches to gait recognition.

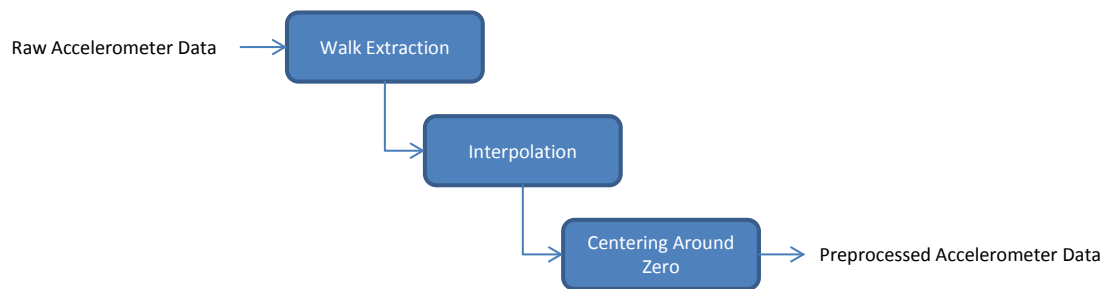


Figure 5.2.: Preprocessing steps.

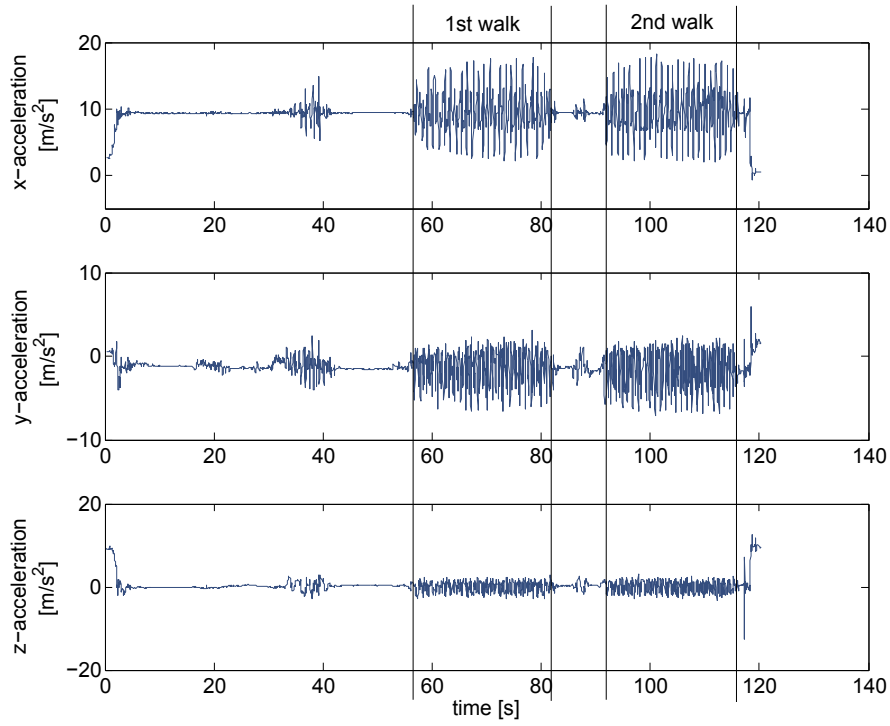


Figure 5.3.: Sample data showing the accelerations measured on the three different axes. Start and end of the two walks are indicated. The x-axis measured the vertical acceleration; therefore, it contains also the gravity.

collections used in this thesis the phones were always positioned in such a way that the vertical acceleration is measured by the x-axis, the forward-backward-acceleration is measured by the y-axis and the z-axis measures the sideways-acceleration. The figure shows that the data contain also parts where the subject was not walking. Because these data should not be part of the evaluation, the *walks* are extracted, containing only data where the subject is walking (indicated in the figure). This is done either purely automatically or the automatic extraction is followed by a visual inspection, and the start and end of the walk are manually corrected if necessary. During automatic extraction, the first step is extracting from the signal of interest the data between the first and the last data value above the mean of the walk to delete the parts at the end and/or beginning containing very low acceleration values because the phone is in a horizontal position before placing it in or after taking it out of the pouch. Afterwards, a convolution filter is applied to the absolute value of the differences of consecutive acceleration values to emphasize the difference of the high acceleration parts (walks) to non-walking sections. Based on this signal the walks are extracted.

Interpolation The data are linearly interpolated to a fixed sampling rate S . The collected data do not have such, due to a limitation of the Android Application Programming Interface (API). Acceleration values can only be obtained when the event `onSensorChanged` is triggered by the system, which is potentially delayed by background processes. The mean sampling rate of the databases used in this thesis is around 42 (for the G1 phone) or around 127 (for the Motorola Milestone) data values per second. Interpolation to different sampling rates (upsampling and downsampling) is evaluated. The reason for upsampling is to prevent loss of too many data values. E. g. peaks might disappear if their original time stamp is between two time values used for interpolation. Downsampling has the advantage of data reduction.

Centering around Zero As the mobile phone is not well calibrated, the acceleration it measures in a stable position (no movement) is not exactly zero or gravity, as it should be in vertical direction, and it is not stable over time. To reduce the influence of this phenomenon, the data are centered around zero. This is done by subtracting the mean μ of the acceleration values of the walk from the data: $\bar{s}_a(t) = s_a(t) - \mu_a$, $a \in \{x, y, z\}$.

5.2. Segmentation

The next step in the feature extraction process is the segmentation. This can be conducted in two different ways. Either the segments are based on gait cycles (see Section 3.1) or are determined by using fixed-length time windows.

5.2.1. Cycle-based Segmentation

A robust cycle extraction method is necessary to handle the circumstances during the performed data collections. Because of using mobile phones, lower and irregular sampling rates were obtained. Also the phone was inside a pouch which was attached to the belt of the subject. This fastening was not as fixed as the one of the dedicated accelerometers mainly used in literature. These factors resulted in data which are in parts highly irregular. Therefore, former cycle extraction methods which are mainly minimum (maximum) based fail when applied to smartphone data, as not every walk does have distinctive minima (maxima). This method overcomes this problem by adapting the process depending on the course of the specific data.

The following subsections describe the conducted process which is illustrated in Figure 5.4. Only the vertical acceleration was used for processing, as it showed the best results.



Figure 5.4.: Flow diagram of the cycle extraction process.

Estimation of Cycle Length To check the quality of the following cycle detection and to adapt this process, it is necessary to estimate the cycle length in advance. This is done by computing the min-salience vector [90]. It contains one entry for each data value. This entry is the number of data values which are between that data value and the following smaller one. This results in high entries in the salience vector in case the corresponding entry in the data vector is a significant minimum. Each entry of the min-salience vector which is greater than the minimal height H_e and has a distance of at least D_e to the next peak is assumed to correspond to a cycle start. The rounded mean value of the distances between neighbored peaks is the estimated cycle length.

Cycle Detection Cycle detection is also based on the salience vector. In many publications only the minima are used to identify a cycle start. Problems occur in case the minimum at cycle start is not greater than the minimum inside the cycle. This mainly happens because the minima are not distinct enough. In these cases there are often distinct maxima. See Figure 5.5 for an example and the corresponding max-salience vector (this is created similarly to the min-salience vector but using the maxima). These maxima occur at the beginning of the cycle and can be used to determine the correct minimum (cycle start).

The proposed method exploits this observation and computes the min-salience vector as well as the max-salience vector. The peaks of the salience vectors which have a minimum height of H_d and a minimum distance of half the estimated cycle length are computed. In case of the min-salience vector these are used as initial cycle starts. In case of the max-salience vector the significant minimum before the maximum has to be determined. To find out if minima or maxima are more suited for calculation of cycle starts, the number of unusual long cycles (with length greater than U_d plus the estimated cycle length) is computed. The version which resulted in less irregular cycles is assumed to be the better one and the respective cycle starts are used.

The previously identified too long cycles, are further divided using the salience vector which has not been used before. E.g. when the initial cycles were computed using the min-salience vector, the max-salience vector is used for dividing the too long cycles. The identified additional cycle starts and the initial ones produce the final set of cycle starts.

Normalize Cycle Length The extracted cycles are of different length. Because some of the distance functions, e. g. Euclidean distance, require feature vectors of same length as input, a normalization of cycle length is necessary if these distances are used in the classification process. Other distances, such as DTW [100], can calculate the distance between cycles of unequal length. Hence, the length normalization is an optional step.

Omit Unusual Cycles The previously identified cycles are cleaned by deleting unusual cycles. The pairwise distance between all cycles is computed using DTW. Those cycles which have a

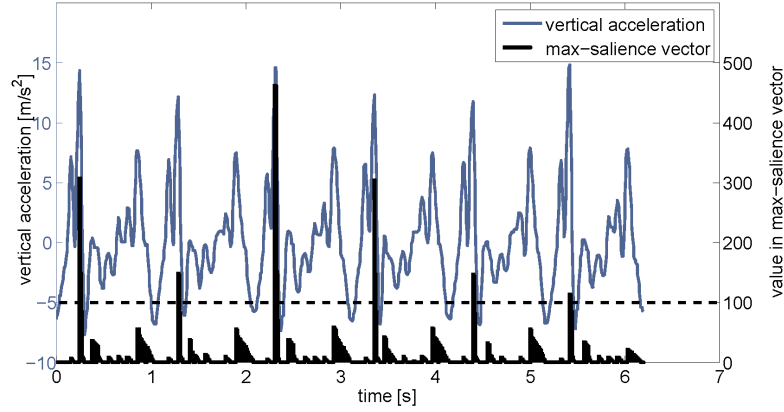


Figure 5.5.: Example for gait cycles without distinct minima and the corresponding max-salience vector. Acceleration data are interpolated to a sampling rate of 120 Hz and centered around zero. The dashed line indicates a sample minimal height H_d used for determining the significant maxima.

distance of at least D_o to at least half of the other cycles are deleted. Depending on the evaluation, this might be the last step of the cycle extraction. The remaining cycles are stored and later used as reference or probe cycles.

Determine Typical Cycle The remaining cycles from the previous step can be further analyzed to determine the most typical cycle of the respective subject. As before, the distances between all remaining cycles are computed. The cycle for which the sum of the distance is minimal is the one which is most typical for the walk and hence is stored for usage as reference or probe cycle.

5.2.2. Fixed-length Time Segmentation

Alternatively to the cycle-based segmentation, the data can be divided into segments of a fixed time length without considering the course of the signal. Segment lengths between 2 and 7.5 s are used in this thesis, the segments overlap by 50%.

5.3. Feature Extraction

Feature extraction is an optional step. Instead, it is also possible to directly use the raw data contained in one segment. Features are extracted from the x-, y- and z-direction as well as from the magnitude vector $\bar{s}_m = \sqrt{\bar{s}_x^2 + \bar{s}_y^2 + \bar{s}_z^2}$.

For each fixed-length time segment and acceleration-direction one feature vector is extracted, containing one or more of the following features:

Mean Mean value of the segment.

Min Minimal value of the segment.

Max Maximal value of the segment.

Std Standard deviation of the segment.

Diff Difference of maximal and minimal value of the segment.

Bin Relative histogram distribution in linearly spaced bins between the minimum and the maximum acceleration in the segment. Five and ten bins were used.

RMS Square root of the mean of the squares of the acceleration values of the segment:

$$\text{rms} = \sqrt{\frac{\bar{s}_a^2(1) + \bar{s}_a^2(2) + \dots + \bar{s}_a^2(n)}{n}}, \text{ where } n \text{ is the number of data values in the segment, } a \in \{x, y, z, m\}.$$

Cross Number of sign changes in the segment.

MFCC Mel-frequency cepstral coefficients, settings see below.

BFCC1 Bark-frequency cepstral coefficients, settings see below.

BFCC2 Bark-frequency cepstral coefficients, settings see below.

5.3.1. Mel- and Bark-Frequency Cepstral Coefficients and Variations

MFCCs and Bark Frequency Cepstral Coefficients (BFCCs) are the predominant features in speaker and speech recognition. As will be shown in the evaluation, they are also suitable for gait recognition. In addition to the Mel and Bark scale, a linear scale is evaluated (Linear Frequency Cepstral Coefficient (LFCC)), to evaluate the suitability of these speech-related scales for gait. The feature extraction process is depicted in Figure 5.6 and consists of the following steps [84]¹:

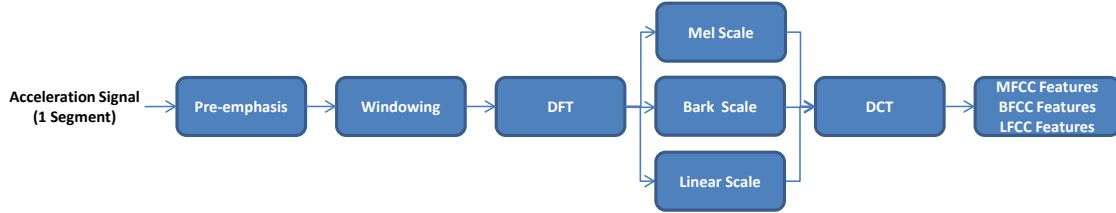


Figure 5.6.: Process to create cepstral coefficients and the variations using a linear scale.

Pre-emphasis A pre-emphasis filter is applied to each segment

$$p(1) = \bar{s}_a(1)$$

$$p(t) = \bar{s}_a(t) - 0.97 * \bar{s}_a(t - 1), \text{ for } t = 2, \dots, \text{segmentLength}; a \in \{x, y, z, m\}.$$

Windowing The segment is further divided into overlapping windows of size *window size*, where one window starts *window hop time* seconds after its predecessor. A Hamming window [116] is used to avoid edge effects.

¹The implementation of D. Ellis is used. <http://labrosa.ee.columbia.edu/matlab/rastamat/>

DFT The Discrete Fourier Transform (DFT) is computed for each window. The number of spectral components depends on the window size.

Scaling To smooth the spectrum, the spectral components of each window are grouped in 40 bins. For regular MFCCs and BFCCs the spacing of the bins is not equal but based on the Mel [135] or Bark [164] scale, depending on the feature. These scales take into account that, for speech, the lower frequencies are perceptually more important than the higher frequencies. Because this might not be the case for gait data, also linear bin spacing is evaluated.

DCT The goal of this step is a decorrelation of the spectral features by transforming the feature space. In the area of speech processing this is not done via PCA, as usual. In contrast, to create cepstral coefficients the Discrete Cosine Transform (DCT) is applied to the logarithm of the amplitude spectrum to obtain 13 cepstral coefficients.

Averaging At this point, 13 coefficients per window exist. For each coefficient the mean over all windows is computed, producing the final coefficients for the segment.

The following settings are chosen for the above stated features MFCC, BFCC1 and BFCC2²

MFCC window size = 1.44 s, window hop time = 0.048 s, max frequency = 10 Hz

MFCC1 window size = 1.12 s, window hop time = 0.032 s, max frequency = 7.5 Hz

BFCC2 window size = 1.92 s, window hop time = 0.048 s, max frequency = 8.75 Hz

These configurations are determined in an optimization process on two different databases. The first database was collected using a Motorola Milestone with a sampling rate of around 125 Hz [23]. The second database is the same as used in Chapter 6 using a G1 phone yielding a sampling rate of around 40 Hz. Features MFCC and BFCC1 were optimized on the first database, BFCC2 was optimized on the second one. For some evaluations the Bark or Mel scale is replaced by a linear scale to validate that the features are the correct choice for gait data. The remaining feature extraction process and configuration is kept the same.

5.3.2. Feature Selection Using Discriminative Potential Score

During the evaluations the single features will be combined to feature vectors. The selection of features which compose the feature vectors is done in two different ways. Mainly, the performance of the single features is evaluated and best performing features are combined. Alternatively, features are selected based on their Discriminative Potential Score (DPS). This score was introduced by Mráček et al. [99]. In their work, anatomical features of 3D face images are computed and the most suitable features are determined based on their DPS.

The idea is to identify features which have a low intra-class variability and a high inter-class variability. For features consisting of multiple entries (like the cepstral coefficients) each entry is considered as separate feature. The values of each feature have to be normalized to the range [0, 1]. Afterwards, for each subject i and feature j the value range v_{ij} is computed. From these values the following statistics are extracted:

Mean Deviation μ_j mean of the value ranges.

²The given parameter settings are given w.r.t. the gait frequency range of 0-10 Hz [156]. For input to the implementation of Ellis [44] they are adjusted.

Standard Deviation s_j mean standard deviation of the feature over all subjects.

Maximum Deviation m_j maximum of the value ranges.

These values are used to identify a low intra-class variability. It is assumed that feature values of all subjects have to be uniformly distributed in order to yield a high inter-class variability. Therefore, the value c_j is determined as the correlation of the feature values to the uniform probability density function.

Based on these values the DPS of feature j is calculated as follows:

$$\text{dps}_j = (1 - (\mu_j + s_j + m_j)) + c_j.$$

Based on this score, different feature vectors can be created by setting a threshold for the minimal required DPS. Using all other previously described features results in a feature vector of length 244. By setting the thresholds to DPS values between -0.4 and 0.7 , only a limited amount of features remains. The thresholds are chosen to evaluate low and high reductions of the feature vector length, and result in feature vectors of length between 241 and 12 entries. The features contained in the different feature vectors are listed in Appendix H.

5.4. Classification

As mentioned earlier, two different approaches for classification exist: template-based classification and classification via machine learning algorithms.

5.4.1. Template-based Classification

During template-based classification, distances between the stored reference and the probe have to be computed. The following distances are used in this thesis to calculate the difference between two cycles, where l_1 is the length of the reference cycle r and l_2 is the length of the probe cycle p .

Euclidean Distance The cycles have to be normalized in length for this distance, which is computed as follows:

$$\text{Euc}(r, p) = \sqrt{\sum_{j=1}^{l_1} (r(j) - p(j))^2}.$$

DTW³ A cost matrix C is computed, which contains the squared differences between the data values of the two compared cycles. Hence, the cost for substituting an entry is the square of the difference of these two values. Similar, the cost for deleting or adding a value is the squared difference of the previous cycle value and the one to be deleted or added, respectively. Using this cost matrix, the distance matrix D is created in the following way:

$$\begin{aligned} D(1, 1) &= C(1, 1) \\ D(m, 1) &= C(m, 1) + D(m - 1, 1) \\ D(1, n) &= C(1, n) + D(1, n - 1) \\ D(m, n) &= C(m, n) + \min(D(m - 1, n), D(m - 1, n - 1), D(m, n - 1)), \end{aligned}$$

³The implementation of T. Fetty published at Matlab Central is used.

<http://www.mathworks.com/matlabcentral/fileexchange/6516-dynamic-time-warping>

for $m = 2, \dots, l_1$ and $n = 2, \dots, l_2$. This construction assures that the lower right entry of D contains the final distance between the two compared cycles: $\text{DTW}(r, p) = D(l_1, l_2)$.

Rotation Dynamic Time Warping (RDTW) As a first step, the minima of the probe cycle are computed. The cycle is rotated, such that each of these minima is once the cycle start. For each rotated cycle p_i , the DTW distance is computed. The minimum of these distances is the final distance between the two cycles: $\text{RTDW}(r, p) = \min_i(\text{DTW}(r, p_i))$, where $i = \{1, \dots, n\}$ and n is the number of computed minima in the probe cycle.

Cyclic Rotation Metric (CRM)⁴ For this distance, the cycles have to be normalized in length, i.e. $l_1 = l_2$. The probe cycle is rotated l_1 times. For each rotated cycle the Manhattan distance is calculated. The minimal Manhattan distance together with the DTW distance of the corresponding cycles is the final distance pair. Two thresholds are applied in the classification process, one for each distance. At least one of them has to be lower than the corresponding threshold.

Depending on the number of reference and probe cycles, various distance values exist for the authentication of one subject, and the authentication decision has to be based on these values. The simplest case is the one, where only one distance exists. If this distance is below a pre-selected threshold a match is obtained, otherwise a non-match. If multiple reference cycles exist, the results are combined to get one authentication result per probe cycle. The combined result is a match if at least half of the single results are a match. When k probe cycles are used for authentication, after this step, there are still k authentication results. These are input to a voting mechanism as described in Section 5.5.

5.4.2. Machine Learning Classification

Three different machine learning algorithms for classification were evaluated and identified to deliver suitable results for accelerometer-based gait recognition, namely HMMs, SVMs, and the k-NN algorithm. These algorithms are shortly presented in the following sections. All of these methods have in common, that they need to be trained during enrolment. The trained classifiers are stored in the database and serve as references.

5.4.2.1. Hidden Markov Models

HMMs were introduced during the mid-60's by Baum et al. [11] and are commonly used for biometric authentication tasks, e.g. in speaker [101] or writer recognition [128]. In [122], Rabiner and Juang give the definition of HMMs as follows:

“An HMM is a doubly stochastic process with an underlying stochastic process that is *not* observable (it is hidden), but can only be observed through another set of stochastic processes that produce the sequence of observed symbols.”

Formally, an HMM is a 3-tuple $\lambda = (A, B, \pi)$. For an HMM with N states and observation sequence $O = o_1, o_2, \dots, o_T$, we have the following definitions:

⁴The implementation of M. O. Derawi is used [41].

State Transition Probabilities Matrix $A = \{a_{ij}\}$ contains the probabilities, that, being at time t in state s_i , the following state is s_j ($i, j = 1, \dots, N$). In general, it is possible to stay at the same state ($i = j$).

$$a_{ij} = P(X_{t+1} = s_j | X_t = s_i), \text{ where}$$

$$a_{ij} \geq 0, \sum_{j=1}^N a_{ij} = 1 \quad \forall i.$$

Observation Symbol Probabilities Set $B = \{b_i(k)\}$ is the set of probabilities of observing symbol $k \in K$ at a given state s_i , $i = 1, \dots, N$:

$$b_i(k) = P(o_t = k | X_t = s_i).$$

In this thesis, continuous HMMs are applied. Therefore, the symbol space K is replaced by \mathbb{R} and the observation probability is defined by a Gaussian Mixture Density [67].

Initial State Distribution $\Pi = \{\pi_i\}$, $i = 1, \dots, N$ contains the initial state distribution:

$$\pi_i = P(X_1 = s_i)$$

Figure 5.7 shows a 3-state left-right HMM as mainly used in this thesis. The transition probability from one state to the previous one equals zero. The first and last state are non-emitting.

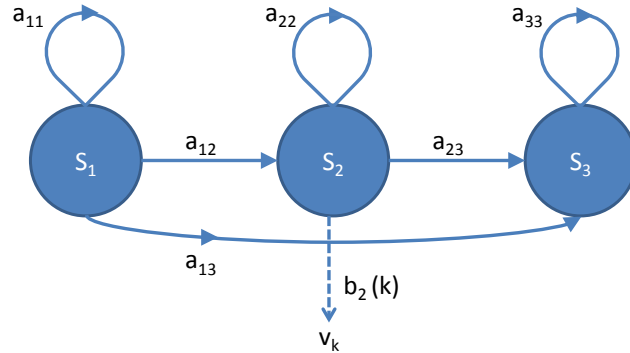


Figure 5.7.: 3-state left-right HMM.

The following three key problems are connected to HMMs [122]:

Problem 1 How do we compute $P(O|\lambda)$, which is the probability that the observation sequence $O = o_1, o_2, \dots, o_\tau$ is obtained from the model $\lambda = (A, B, \pi)$.

Problem 2 How do we determine the optimal state sequence $I = i_1, i_2, \dots, i_\tau$ for a given observation sequence $O = O_1, O_2, \dots, O_\tau$.

Problem 3 How do we adjust the parameters of model $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$.

In this work, only Problem 1 and 3 are relevant. Problem 3 is also called *Training Problem* as it focuses on identifying suitable parameters of the HMM which represent the training data best. This is done in an iterative process using the Baum-Welch algorithm [122]. Problem 1, the *Evaluation Problem* has to be solved during authentication. Given a trained HMM and probe data, the probability that the given probe data are represented by the model is computed using the Viterbi algorithm [151].

In this thesis the *world model approach* is applied, which was first introduced by Carey et al. in 1991 [28] and has become the predominate approach in speaker verification systems [16]. In this approach two HMMs are considered in each test iteration. One is the genuine model λ_{gm} , which is trained during enrolment with data of the genuine user, and the other is the world model λ_{wm} . This world model (also called general model or universal background model) is trained using data from a large amount of different subjects. Because this model is independent of λ_{gm} , it can be trained before enrolment of a specific subject. For evaluations in this thesis the Hidden Markov Model Toolkit (HTK) [161] was used.

Enrolment Enrolment focuses on Problem 3, where the HMM λ_{gm} for the genuine user has to be adapted to the training data. The training data consist of several feature vectors of that user.

Authentication Authentication corresponds to Problem 1. Using the Viterbi algorithm, we obtain for each model, λ_{gm} and λ_{wm} the probability that the given probe data are represented by the model. Classification is based on the difference of these two probabilities and a threshold.

5.4.2.2. Support Vector Machines

SVMs are designed to solve binary classification problems. This makes them suitable for biometric verification applications, where *genuine* and *impostor* data have to be classified. The general idea of SVMs is to map the data to a higher dimensional space, where it is linearly separable. A hyperplane is determined which, in the ideal case, separates the data of the two classes. In the following, a brief derivation of the concept is given. A more detailed description can be found in [15] and [27]. The experiments were conducted applying the SVM implementation LIBSVM [32].

The separating hyperplane is defined as $H = \{x \mid \langle w, x \rangle + b = 0\}$, where w is a normal to the hyperplane. The distance of the plane to the origin is $\frac{|b|}{\|w\|}$, where $\|w\|$ is the Euclidean norm of w . During training, the SVM determines the hyperplane with the largest margin. The margin is limited by the two hyperplanes $H_1 = \{x \mid \langle w, x \rangle + b = 1\}$ and $H_2 = \{x \mid \langle w, x \rangle + b = -1\}$, hence, it is of size $\frac{2}{\|w\|}$. Those points lying on H_1 or H_2 are called support vectors. Figure 5.8 depicts the situation.

To determine the hyperplane with the largest margin the following optimization problem is solved:

$$\min \frac{1}{2} \|w\|^2 \quad (5.1)$$

$$\text{subject to } y_i(\langle w, x_i \rangle + b) \geq 1 \forall i.$$

$y_i \in \{-1, 1\}$ specifies the class corresponding to training vector i , $i \in \{1, \dots, m\}$, where m is the number of training vectors. To generalize the problem to be able to handle data which is non-linearly separable, the Lagrange formulation of the optimization problem has to be considered.

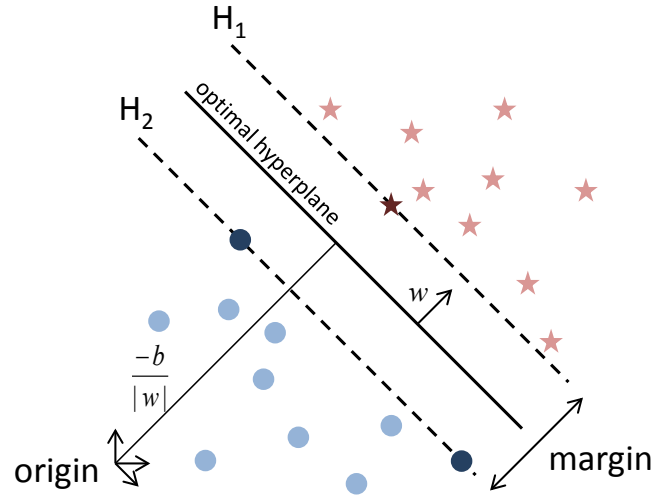


Figure 5.8.: Hyperplane separating the two classes (stars and circles). Support vectors are darkened. (Based on [27].)

The Lagrangian dual of equation 5.1 results in the following dual problem

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \quad (5.2)$$

$$\text{subject to } \sum_{i=1}^m y_i \alpha_i = 0 \text{ and } \alpha_i \geq 0, i = 1, \dots, m.$$

The variables α_i are the Lagrange multipliers. Only the support vectors have non-zero Lagrange multipliers and define the position of the planes.

Now, two supplementary properties are added to SVMs. First, a cost parameter C is introduced. This cost parameter can be seen as a penalty given to data values lying on the wrong side of the hyperplane (classification errors). This changes the Lagrangian dual to

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \quad (5.3)$$

$$\text{subject to } \sum_{i=1}^m y_i \alpha_i = 0 \text{ and } C \geq \alpha_i \geq 0, i = 1, \dots, m.$$

Second, the data are mapped to a higher dimensional space to improve the separability of the training data. This idea is based on work by Cover [34], who showed that in dimension d the expected maximum number of vectors, randomly assigned to a class, which can be linearly separated is $2d$. The mapping to a higher dimensional space is performed via a kernel function

$K(x_i, x_j) \equiv \phi(x_i)^T \phi(x_j)$. The adapted Lagrangian dual problem is

$$\begin{aligned} \max_{\alpha} \quad & \sum_i \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m y_i y_j \alpha_i \alpha_j \langle \phi(x_i), \phi(x_j) \rangle \\ \text{subject to} \quad & \sum_{i=1}^m y_i \alpha_i = 0 \text{ and } C \geq \alpha_i \geq 0, i = 1, \dots, m. \end{aligned} \quad (5.4)$$

Solving this problem yields the normal $w = \sum_{i=1}^m y_i \alpha_i \phi(x_i)$ to the plane. From this, distance b can be recovered by the support vectors (for which holds $\alpha_i > 0$).

Enrolment During enrolment, the SVM has to be trained. Input to the machines are d -dimensional feature vectors $x_i \in R^d, i = 1, \dots, m$ and their corresponding labels $y_i \in \{-1, +1\}$. Genuine data are labeled as "1", data of several other subjects are used to represent class "1".

This study uses the Gaussian Radial Basis Function (RBF) kernel $K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2)$, where $\gamma > 0$ is an adjustable kernel parameter. For a given cost parameter C and kernel parameter γ the dual quadratic problem 5.4 is solved for the training data. The support vectors s_v , the corresponding obtained $y_v \alpha_v$, b as well as the label names and kernel information are stored for classification ($v = 1, \dots, N_s$, where N_s is the number of support vectors s_i).

The optimal cost parameter C and the kernel parameter γ are determined during enrolment via a grid search. This means that for each combination of C and γ , using a defined range and grid size, FMR and FNMR are determined via cross-validation.

Authentication Given a trained model and the probe data p , the classification is based on the decision function

$$f(p) = \text{sign}(\sum_{v=1}^{N_s} \alpha_v y_v K(s_v, p) + b),$$

where N_s is the number of support vectors s_i .

5.4.2.3. k-Nearest Neighbor Algorithm

The k-NN algorithm is a simple Instance Based Learning (IBL) algorithm [5]. For these algorithms, the stored model consists of a set of instances (feature vectors) together with their corresponding class labels and the used distance function. The Euclidean distance is applied, which is most common. During classification, the distances between the probe vector and the stored reference instances are computed. The class labels of the k closest vectors are analyzed. The class of the majority of the k neighbors is assigned to the probe vector. In case no majority exists, the genuine class is assigned. See Figure 5.9 for a schematic presentation of the concept. The implementation of the Waikato Environment for Knowledge Analysis (WEKA) library is used [66] having the advantage that a port to the Android platform exists.

Enrolment The feature vector extracted from the genuine data recorded during enrolment, the stored impostor instances and the corresponding class labels constitute the model.

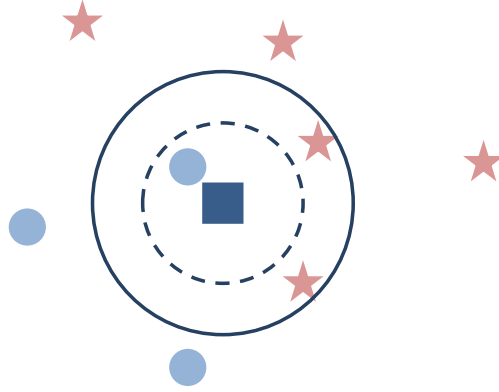


Figure 5.9.: Example for k-NN classification. For $k=1$ the circle class is assigned to the square. For $k=3$, it is classified as belonging to the star class.

Authentication The k training instances which are nearest to the given probe data are determined. If at least half of these k neighbors belong to the genuine class, this class is assigned to the probe data⁵.

5.5. Voting

Independent of the applied feature extraction and classification method, the classification results can be further processed. Currently, one classification result is obtained per walk or per segment; hence it is based on a very short walking duration of less than 30 seconds. To get more reliable results, one can group several consecutive classification results and convert them to a single result. The advantage is that short disturbances in the gait, e. g. due to curbs or slipping, are compensated. Figure 5.10 depicts the applied approach, called *quorum voting*.

The idea is inspired by a petition quorum. Each single classification result votes either for genuine or for impostor. A defined number of $\#V$ votes is analyzed and the number of votes for genuine $\#V_g$ is computed. If this number exceeds the pre-selected threshold $\#GV$ the whole quorum votes for genuine, otherwise for impostor. If a low threshold is chosen, a higher weight is given to the votes for genuine. This is decreasing the FNMR while increasing the FMR. The evaluation showed that the decrease of FNMR is stronger than the increase of FMR, resulting in a lower EER or HTER. Depending on the group size $\#V$ and the amount of available probe data, one or more groups can be created and used to compute the overall error rates. To obtain results which are not influenced by the amount of available groups, only the first group of each probe subject is considered in the evaluation.

⁵Alternatively, decision could incorporate weighted distances.

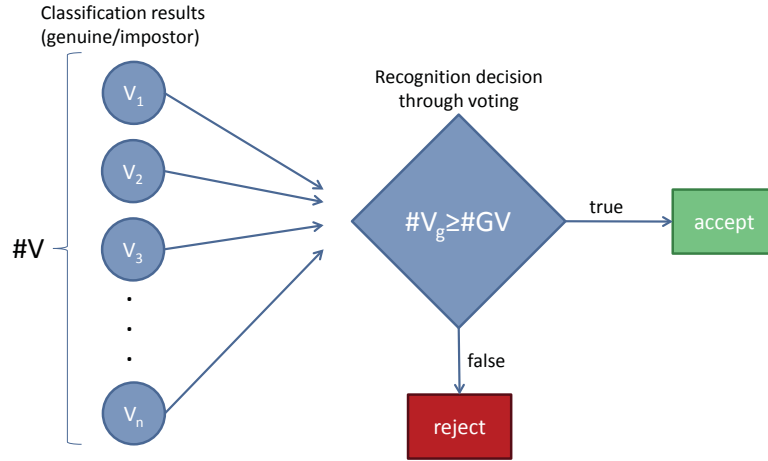


Figure 5.10.: Quorum voting scheme.

5.6. Summary

The previous sections introduced the different components which constitute the contributed gait recognition methods. Identical for all methods are the preprocessing steps, including extraction of the walk data, interpolation to a fixed sampling rate, and centering around zero. The segmentation is either based on gait cycles or using fixed-length time segments. Several features are presented which can be extracted from the segments. The created feature vectors can either be compared via template matching or classified using machine learning algorithms.

For the cycle-based segmentation, a new approach is presented. MFCCs and BFCCs belong to the selected features. This was encouraged by their success in speaker recognition and based on the observation of the similarity between gait and speech signals. Never before, these have been applied to gait data. These features are varied by applying a linear scale instead of the Bark or Mel scale.

A voting scheme is presented which helps to compensate small disturbances in the gait, e.g. due to uneven ground, by incorporating several classification results into the final authentication result.

So far, the selected machine learning algorithms have not been thoroughly or not been at all evaluated for biometric gait recognition. The success of the developed methods is confirmed in Chapter 6, 7, and 8, which contain a thorough evaluation on three different gait databases. A comparison of the results is given in Chapter 9.

CHAPTER 6

BENCHMARKING DIFFERENT ACCELEROMETER-BASED BIOMETRIC GAIT RECOGNITION METHODS

This chapter describes the initial evaluation of HMMs and SVMs, applied to accelerometer-based gait data as described in Chapter 5. In total three different data collections were conducted and are used for evaluation in this thesis. The results in this chapter are obtained using the initial data collection, which is described in the following section. The evaluation of a cycle extraction method on the same database is presented in [41]. In this chapter, it is evaluated if HMMs and SVMs yield a biometric performance which is competitive to the one obtained applying the cycle extraction method, which is the basic approach for accelerometer-based biometric gait recognition. In addition, suitable feature sets are identified. This chapter can be seen as a pre-evaluation targeting at the first four research questions concerning classification using machine learning algorithms, identification of suitable features, necessary amount of training data, and aging of template data. It is based on the publications [106] and [109].

6.1. Initial Data Collection Using a G1 Phone

The data is collected using a standard G1 mobile phone [64]. The integrated accelerometer is a AK8976A developed by Asahi Kasei Microsystems [10]. The stated power consumption is 6.7 mA during sensor operation and an average of $460\mu\text{A}$ with measurements at 100 ms intervals. The G1 uses the Android platform and a software was written to access the accelerometer and output the data from the sensor to a file (40-50 samples per second for each of the three directions x, y and z). While recording the gait data, the phone has been placed in a pouch attached to the belt or trousers of the subject on the right-hand side of the hip. The phone is positioned horizontal, the screen points to the body, the upper part of the phone points in walking direction (see Figure 6.1).

Subjects had to walk about 37 meters down the hall at CASED on flat carpet (see Figure 6.2). At the end of the hall the subjects had to wait two seconds, turn around, wait again and then walk back the same distance. No fixed pace was given, so the walking speed varies from subject to subject depending on what is sensed to be normal.

Full data sets of 48 volunteers were collected (see Table 6.1 for age and gender distribution).



Figure 6.1.: Phone attached to subject and the three axes in which acceleration is measured.



Figure 6.2.: Photograph of the indoor walking setting.

Each of them did two sessions at two different days wearing his normal shoes. From the data collected on each day, two walks could be extracted (see Figure 5.3). One, when the subject was walking in one direction and the other one when he was walking back. So in total there are four walks for each subject.

6.2. Hidden Markov Models

In this section HMMs are applied to the database, which is a new approach for accelerometer-based gait recognition. The presented results were published in [109]. No feature extraction was performed in this evaluation, but the accelerometer data was directly used for training and classification.

The data used for the experiments consists of four walks for each of the 48 subjects. These walks were interpolated to a fixed sampling rate of 200 samples per second and further divided into parts of 3 seconds each which resulted in a total of 1651 walk sections. This configuration has been empirically determined to deliver best results. Depending on the speed of a subject's

	< 20	20 – 24	25 – 30	> 30	unknown	total
male	1	1	25	9	2	38
female	0	5	4	0	1	10
total	1	6	29	9	3	48

Table 6.1.: Age and gender distribution of volunteers of the first data collection.

walk this resulted in 28 to 37 data sets (walk sections) per subject. In order to make sure that all the subjects have an equal number of walk sections, 28 walk sections per subject (total of 1344 walk section for 48 subjects) were used in the evaluation. The accelerations in x-, y- and z-direction had to be converted to HTK format. This conversion included transforming the data into a matrix with 26 columns which contains all x-values in the first rows, followed by all y-values and all z-values. It was paid attention that no row contains data of two different directions.

6.2.1. Evaluation

The goal of the analysis is to know how often the enrolled user is not recognized by his phone and how often a foreign subject is recognized as the enrolled user, resulting in unauthorized access. Therefore, 48 tests were executed and in each test a different subject was chosen to be the genuine user. This subject should be recognized by the system as the authorized subject and the HMM needs to be trained accordingly. All the other subjects should be rejected (identified as belonging to the impostors) by the system. Following the world model approach, for each of these two groups one left-right-HMM was created. Different tests using five to eight states and four to six Gaussian mixtures per emitting state were done. For training these two HMMs, the first 20 walk sections of the genuine user and all 840 walk sections of 30 subjects from the impostor group were used. The remaining walk sections of the genuine and the impostor group were left for testing purposes. This process was repeated by picking a different subject as genuine user in each run. For each chosen genuine user this results in a different number of false matches (a subject of the impostor group was identified as genuine) and false non matches (the genuine user was not recognized as such) as output of the Viterbi algorithm (see Section 5.4.2.2). On a Intel(R) Core(TM)2 Duo CPU P9500@2.53GHz, 2.53Hz with 4GB RAM the test for one subject for a seven state HMM (5 mixtures) took on average 0.73 seconds (standard deviation was 0.22).

In order to get realistic results with respect to the application scenario, majority voting was applied for the calculation of error rates, which is a special form of the voting described in Section 5.5. All sections of one walk are grouped; hence, the number of sections in the walk corresponds to $\#V$. Only if more than half of its walk sections is rejected, it is considered as a false reject, meaning that $\#G = \#V/2$.

Table 6.2 shows the results obtained when using different number of states (ranging from five to eight) and different numbers of mixtures per state (ranging from four to six). One can see that the 7-state HMM always gives best results in terms of HTER.

In the previously stated results, all data which was not used for training was used for testing. Therefore, for the genuine test the majority voting is based on 24 seconds of data (8 sections of 3 seconds each) and for the impostor test the majority voting is based on 84 seconds of data (28 sections of 3 seconds each). The results shown in Table 6.3 were obtained when using eight walk sections (i.e. 24 seconds) of both the genuine and the subjects belonging to the impostor group for testing. Only the results of the best performing 7-state HMM are listed.

States	Mixtures	FMR	FNMR	HTER
5	4	5.76%	27.08%	16.42
5	5	4.53%	16.67%	10.60
5	6	4.17%	14.58%	9.38
6	4	12.38%	8.33%	10.36
6	5	6.37%	14.58%	10.48
6	6	3.92%	12.50%	8.21
7	4	12.38%	2.08%	7.23
7	5	6.62%	10.42%	8.52
7	6	2.70%	12.50%	7.60
8	4	6.62%	22.92%	14.77
8	5	3.80%	18.75%	11.28
8	6	1.35%	29.17%	15.26

Table 6.2.: Results obtained for different numbers of states and Gaussian mixtures using all test data.

No. of Mixtures	FMR	FNMR	HTER
4	15.69%	2.08%	8.89
5	10.29%	10.42%	10.36
6	4.78%	12.50%	8.64

Table 6.3.: Results obtained with a 7-state HMM and different numbers of Gaussian mixtures when using 24 seconds of test data.

From these results, it can be seen that when changing the number of mixtures, the HTER stays nearly the same but the fractions of FMR and FNMR change. Hence, the optimal number of mixture components can be chosen based on the usability (i.e. lower FNMR) and security (i.e. lower FMR) requirements of the system.

When comparing the FMR in Table 6.2 and 6.3 one can see that it is decreasing when the results are based on a longer time period (i.e. a higher amount of walk sections). This decrease is further visualized in Figure 6.3. Due to the small size of the database, the concurrent decrease of FNMR cannot be confirmed in this evaluation. The following evaluations on larger databases involve the identification of the most suitable amount of test data by applying the more flexible voting method introduced in Section 5.5.

6.3. Support Vector Machines

SVMs are the second machine learning approach which was evaluated on the database. The results presented in this section have been published in [106]. Data was interpolated to sampling rates of 50, 100, and 200 Hz. Features were extracted from segments of lengths 5, 7.5, and 10 s.

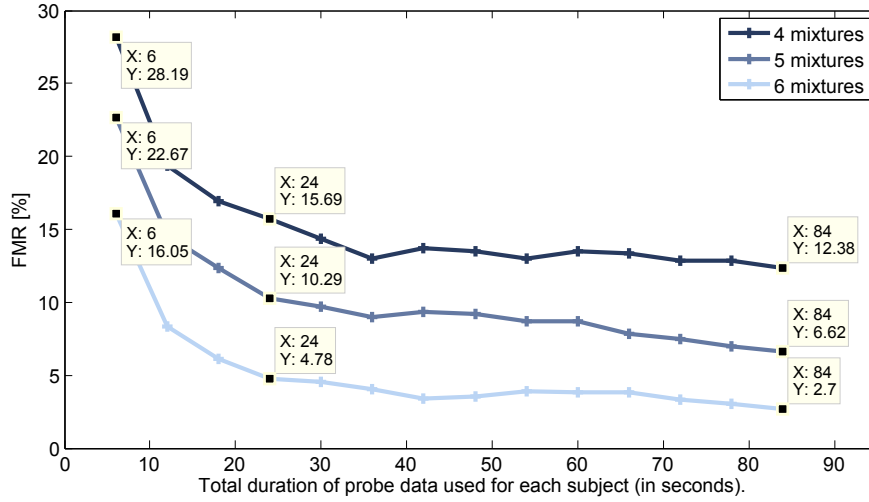


Figure 6.3.: FMR depending on the length of the probe data (7-state HMM). The boxes show the minimal and maximal FMR as well as the previously stated results obtained when using 24 seconds of probe data (see Table 6.3).

6.3.1. Evaluation

In a first step, the discrimination capabilities of single features and of combined features were evaluated. To analyze the influence of the different acceleration directions, the three axes were evaluated separately and combined. Afterwards the optimal sampling rate and segment length were determined, and the error rates were further reduced by applying the voting method.

Identification of Best Feature Set

To obtain realistic cross-day results, training was performed using acceleration data from the first day, and the trained models were tested using data from the second day. For preprocessing, a sampling rate of 100 Hz and a segment length of 7.5 s were selected. Features (see Section 5.3) are extracted from the segments. FMR and FNMR were determined via cross-validation.

Feature vectors for the three different axes and the magnitude vector are evaluated separately and combined. The feature length for all statistical features is 1, except for the binned distribution where five bins are used, resulting in a feature vector length of 5. As described in Section 5.3.1, the MFCC and BFCC2-features consist of 13 elements each. Of course, when features of the different axes are combined, the length is four times the single length. From the results given in Table 6.4 it is apparent that the best performance is obtained when all three axes and the magnitude vector are combined.

In a second step, various combinations of the features are evaluated. Table 6.5 presents the results. The first four feature sets are created by taking the best single feature BFCC2_xyzm and adding the second best, third best etc. MFCCs are deleted from the feature sets to obtain set 5 to 7. As these results are better, further combinations of BFCC2 and several statistical features were tested (set 8 to 11). The obtained results are similar to the ones when combining BFCC2 with the best performing statistical features. Overall, feature set 7 yields the lowest HTER.

Feature	X		Y		Z		Magn.		Combined	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
Max	0.5%	96.9%	0.5%	98.2%	1.9%	96.7%	1.0%	97.5%	4.2%	82.1%
Min	0.5%	99.3%	0.9%	97.5%	0.4%	97.8%	1.2%	98.2%	4.7%	77.4%
Mean	0.0%	99.8%	1.1%	99.1%	0.2%	99.5%	0.5%	98.4%	3.6%	89.4%
Std	0.3%	97.5%	0.6%	96.0%	0.3%	100.0%	0.2%	98.7%	4.2%	81.0%
RMS	0.9%	96.2%	0.6%	95.4%	0.3%	100.0%	0.5%	97.4%	4.6%	86.7%
Cross	0.2%	99.3%	0.1%	99.1%	0.6%	99.5%	0.0%	100.0%	5.3%	85.8%
Bin5	6.2%	87.2%	5.7%	90.0%	4.7%	91.1%	4.5%	91.4%	5.2%	76.8%
MFCC	6.1%	76.3%	3.6%	81.4%	4.1%	82.1%	4.2%	78.8%	1.5%	72.1%
BFCC2	6.6%	74.5%	4.6%	77.9%	3.5%	82.9%	3.8%	76.3%	1.5%	67.9%

Table 6.4.: Evaluation of discrimination capabilities of single features.

Set	Used Features	Length	FMR	FNMR
1	BFCC2, MFCC	104	1.3%	69.3%
2	BFCC2, MFCC, Bin5, Min	128	1.2%	69.0%
3	BFCC2, MFCC, Bin5, Min, Std	132	1.2%	69.0%
4	BFCC2, MFCC, Bin5, Min, Std, Max	136	1.3%	68.8%
5	BFCC2, Bin5, Min	76	1.3%	67.0%
6	BFCC2, Bin5, Min, Std	80	1.5%	66.2%
7	BFCC2, Bin5, Min, Std, Max	84	1.5%	65.3%
8	BFCC2, mean, Max, Min, Std, RMS, Cross, Bin5	96	1.2%	66.8%
9	BFCC2, mean, Max, Min, Std, RMS, Cross	76	0.9%	67.5%
10	BFCC2, mean, Max, Min, Std, RMS	72	1.4%	66.6%
11	BFCC2, Max, Min, Std, RMS	68	1.7%	66.2%

Table 6.5.: Evaluation of combined features' discrimination capabilities.

Optimization of Preprocessing Configuration

The influence of segment length and sampling rate is evaluated for the seven best performing feature sets (set 5 to 11). In addition to the previously used segment length of 7.5 s, segment lengths of 5 and 10 s are evaluated. Table 6.6 shows that the segment length of 10 s performs best. The segment length was not increased further, because it is limited by the duration of one walk. Further increasing the segment length would result in less than three segments per walk and a corresponding low amount of probe data.

Set	5 s		7.5 s		10 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR
5	1.7%	70.3%	1.3%	67.0%	1.2%	64.7%
6	1.8%	70.5%	1.5%	66.2%	1.1%	65.0%
7	1.7%	69.6%	1.5%	65.3%	1.1%	64.7%
8	0.9%	67.8%	1.2%	66.8%	0.9%	64.2%
9	1.1%	66.3%	0.9%	67.5%	1.3%	59.9%
10	1.3%	69.1%	1.4%	66.6%	1.4%	63.6%
11	1.2%	69.3%	1.7%	66.2%	1.3%	62.3%

Table 6.6.: Evaluation results for segments of length 5 s, 7.5 s, and 10 s and an overlap of 50% (sampling rate 100 Hz, without voting).

Table 6.7 shows the results for different sampling rates. In addition to 100 samples per second, the data was interpolated to 50 and 200 samples per second. Sampling rate 50 Hz is worse than the other two, which yield very similar results. The best result is an FNMR of 59.9% at an FMR of 1.3% obtained when using feature set 9, a segment size of 10 s, and a sampling rate of 100 samples per second.

Set	50 Hz		100 Hz		200 Hz	
	FMR	FNMR	FMR	FNMR	FMR	FNMR
5	1.0%	67.3%	1.2%	64.7%	1.1%	64.4%
6	1.2%	65.7%	1.1%	65.0%	1.1%	64.7%
7	1.0%	65.2%	1.1%	64.7%	1.2%	64.4%
8	0.9%	65.4%	0.9%	64.2%	0.8%	63.9%
9	1.1%	63.0%	1.3%	59.9%	1.2%	61.5%
10	1.4%	64.3%	1.4%	63.6%	1.5%	63.6%
11	1.3%	63.3%	1.3%	62.3%	1.5%	62.8%

Table 6.7.: Evaluation results for sampling rates of 50, 100, and 200 samples per second and a segment length of 10 s (without voting).

Voting

Table 6.8 presents the voting results separated by three experiment setups:

Cross-day Data of the first session is used for training, data from the second session is used for testing (as in preliminary evaluation).

Same-day Half of the data of one session is used for training, the other half from the same session is used for testing.

Mixed Half of the data from both sessions is used for training, the other half of the data from both sessions is used for testing.

Set	Cross-day		Same-day		Mixed	
	FMR	FNMR	FMR	FNMR	FMR	FNMR
5	4.7%	35.4%	0.3%	29.2%	3.6%	4.2%
6	4.1%	37.5%	0.5%	14.6%	2.6%	4.2%
7	4.1%	37.5%	0.3%	14.6%	4.6%	4.2%
8	3.2%	37.5%	1.0%	10.4%	4.0%	4.2%
9	5.0%	25.0%	1.0%	16.7%	5.9%	6.3%
10	4.9%	27.1%	1.3%	20.8%	7.0%	6.3%
11	4.7%	25.0%	0.8%	20.8%	6.8%	6.3%

Table 6.8.: SVM gait recognition performance (with voting).

Note that the same feature sets were used as in Table 6.5. To identify the optimal combination of votes $\#V$ and genuine votes $\#GV$ various combinations were evaluated. Requiring $\#GV = 1$ proved to be sufficient, while grouping the maximal amount of segments $\#V$. For the cross-day and mixed setting this results in $\#V = 6$, whereas for the same-day setting $\#V$ was only 3 because of the reduced amount of available probe data.

Analyzing these three setups gives the possibility to evaluate the impact of template aging on the recognition results. The cross-day performance represents the most realistic results, because in real live training and testing data are from different days. Therefore, feature set 9 is considered as the best set, because it performed well in all evaluations and provides one of the best cross-day performances of 5.0% FMR and 25.0% FNMR.

Much lower error rates are obtained when training and testing data are not obtained from different days. The mixed-day setup gives an FNMR of 6.3% at an FMR of 5.9% which shows the suitability of SVMs for classification of gait data. The worse results of the same-day scenario attribute to the lower amount of available training and testing data.

6.4. Benchmark of the Approaches

The results presented in this chapter are not comparable in a completely fair way due to different partitions into training and testing data. In the HMM-evaluation, the training data consisted of all normal walks from the first day and the first walk of the second day. This is to some extent comparable to the mixed-day results obtained for SVMs. In this comparison SVMs slightly outperform HMMs. But because the error rates are on the same level (HTER of 6.10% versus 8.64%) none of the methods is clearly ahead of the other.

In [41] a cycle extraction method was applied to the same data set. Again, a mixed-day scenario was evaluated. Reference cycles were extracted from the first walk collected on the first day, probe cycles were extracted from the remaining three walks. In this scenario an EER of 20.1% was obtained, which is more than twice as high as the SVM and HMM results presented earlier. Hence,

the new approach of applying HMMs and SVMs to accelerometer-based gait recognition is very promising, and the methods are further evaluated in the following chapters.

6.5. Summary and Conclusions

This chapter described the initial evaluations executed using SVMs and HMMs. The methods have been evaluated on the same database which was created using an off-the-shelf smartphone. One result of the evaluations conducted in this chapter is the finding that the database does not contain enough data per subject. Because only two normal walks are collected of each subject in each session, the available training and testing data is strongly limited. This is also confirmed by the result in Section 6.3.1 where the cross-day results are much lower than the same-day results. In both evaluations no time-separation is done between training and testing, but in the cross-day evaluation the double amount of training and testing data than in the same-day approach is available.

The following conclusions can be drawn from the results of this chapter:

- Evaluated on the same database, SVMs and HMMs outperform the cycle-based technique.
- MFCCs and BFCCs are promising features which should be further evaluated.
- Because same-day results vary greatly from the more practical cross-day results, evaluations should mainly focus on cross-day evaluations.
- A larger database is necessary to get well-trained SVMs and HMMs and allow for a cross-day evaluation.

These results point already to possible answers to the first four stated research questions. The evaluation indicates that fixed-length time segmentation and classification via machine learning algorithms outperforms the cycle-based methods. In addition, MFCC and BFCC coefficients have been identified as promising features. It has been shown, that the data available in database 1 is not sufficient to thoroughly train the algorithms. Furthermore, the high differences between same-day and cross-day results indicate that template aging influences the error rates. Still, more-detailed evaluations are necessary and are performed in the following chapters.

The evaluations presented in the previous chapter showed, that classification of fixed-length time segments using machine learning algorithms is a promising alternative to techniques using cycle-based segmentation and template comparison. Because the initial data collection did not contain enough data to properly train these algorithms, a second data collection was conducted, which had the main goal to collect a large amount of data of each participant. This chapter describes the data collection and the conducted evaluations. The advantage of the data set is, that it allows to vary the amount of training data to identify the optimal training duration. The main research question regarded here is: *How much training data are necessary when applying machine learning algorithms to accelerometer-based biometric gait recognition?*

In addition to the previously applied SVMs and HMMs, the k-NN algorithm is used for classification. This evaluation also includes the feature vectors created using DPS. A further evaluation analyzes if a cycle-based segmentation improves the classification results for HMMs and if it is sufficient to use the raw cycle as feature vector. To validate the good performance of the algorithms, they are benchmarked to a state-of-the-art method. Publications [105, 108] and [111] are the basis for this chapter.

7.1. Longer Data Collection Using Motorola Milestone

For database 2, data of 36 subjects were collected on two different days. Age and gender distribution are given in Table 7.1. These two sessions were on average 24 days apart (min = 1, max = 125, median = 10.5). At each session the subjects had to walk up and down the same hallway as in the previous data collection (see Figure 6.2). As before, the data collected between starting to walk and stopping at the other end of the hall is called *walk*. In each session, 12 normal walks, 16 fast walks and again 12 normal walks were collected of each subject, later referred to as setting 1, 2, and 3. This resulted in around 30 minutes of walking data for each subject and a total amount of nearly 20 hours of gait data.

The subjects carried a standard cell phone (Motorola Milestone using Android Operating System (OS)) in a pouch attached to the right hip in the same position as the G1 before. The phone contains a built-in accelerometer developed by STMicroelectronics (LIS331DLH) [136]. It has

a low current consumption of $250\mu A$ and measures accelerations in three directions (x-, y- and z-acceleration). An Android application was written to access the accelerometer data and write it to text files. The mean sampling rate is 127.46 data values per second (min = 109.19, max = 128.88). Table 7.2 shows the median number of resulting segments per setting after segmentation to different segment lengths as well as the mean duration of each walk.

	20-24	25-29	30-39	40-49	50+	total
male	3	18	7	1	0	29
female	1	4	2	0	0	7
total	4	22	9	1	0	36

Table 7.1.: Age and gender distribution of participants of the second data collection.

	Setting		
	1	2	3
walk [s]	27.1	19.9	26.3
2 s	25	18	24
3 s	16	12	15
4 s	12	8	11
5 s	9	6	8
6 s	7	5	7
7.5 s	6	4	5

Table 7.2.: Mean duration of a walk in seconds for each setting and the median number of segments after segmentation to different lengths (overlap 50%).

7.2. Hidden Markov Models

The results presented in this section are partly published in [105]¹. The features presented in Section 5.3 are extracted from each segment, except feature *Cross* and *Bin*, which did not contain enough information to train the models.

7.2.1. Evaluation

The evaluation consisted of several consecutive steps for normal and fast walk. First the discrimination properties of single features were analyzed, the influence of the different axes was evaluated and afterwards the best performing features were combined. The feature sets of these tests identified to be the best were used for further evaluation. As last step the quorum voting is applied to obtain the final error rates.

¹Slight variations are due to a small change in the segmentation algorithm.

Single Features

To analyze the discrimination capabilities of the single features, they were tested separately. The first twelve normal walks which were collected on the first day were used for training and the first twelve normal walks of the second day for testing. Different sampling rates were applied for each segment length. Varying the sampling rate and segment length did not significantly change the error rates. The results for sampling rate 50 and segment length 5 s are given in Table 7.3. These settings are chosen for all remaining results stated in this HMM-section. Results for further segment lengths and sampling rates are given in the Appendix (Table B.1 to B.3).

Feature Type	Feature Length	EER
Max_xyzm	4	32.77%
Min_xyzm	4	30.48%
Mean_xyzm	4	31.82%
Diff_xyzm	4	45.91%
Std_xyzm	4	32.23%
RMS_xyzm	4	32.36%
MFCC_xyzm	52	16.96%
BFCC1_xyzm	52	19.85%
BFCC2_xyzm	52	18.08%

Table 7.3.: Cross-day results for single features when using the best performing sampling rate 50 Hz and segment length 5 s.

MFCC_xyzm is the best single feature, resulting in an EER of 16.96%. When the Mel scale is replaced by a linear scale in the feature extraction process, the EER increases slightly to 17.74%. Hence, the Mel scale is appropriate, but the influence of the applied scale is low. The statistical features yield nearly twice as high EERs.

To evaluate the performance of the different axes and the magnitude vector, MFCCs were computed for each axis separately (for simplicity the magnitude vector, denoted with m , is referred to as axis as well). The results are given in Figure 7.1 as Detection Error Trade Off (DET)-curves. Of the single axes, the x-axis yielded best results, followed by the y-axis and the magnitude vector. Therefore these were combined into one feature vector. In addition, the DET-curve of MFCC_xyzm is displayed, yielding the best results.

The fast walk results (setting 2) for BFCC1, BFCC2, and MFCC are given in Table 7.4 for selected axes. In contrast to the normal walk results, the combination of x-axis and magnitude vector gives best results and BFCC1_xm is the best feature vector.

	BFCC1	BFCC2	MFCC
x	18.12%	18.15%	19.02%
xm	14.37%	15.34%	16.50%
xyzm	19.02%	17.19%	17.44%

Table 7.4.: Fast walk results for single features and different axes (EER). The data are preprocessed using sampling rate 50 Hz and segment length 5 s.

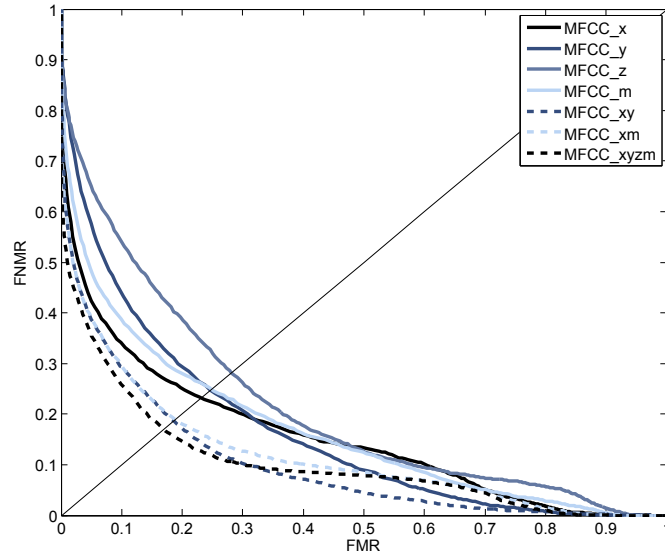


Figure 7.1.: HMM results for feature MFCC and different axes (m = magnitude vector). The data are preprocessed using sampling rate 50 Hz and segment length 5 s.

Combined Features

The results of the single features indicate that MFCC and BFCC outperform the statistical features. One reason might be the much higher length of the feature vector. Therefore, all evaluated statistical features, except the worst performing Diff_xyzm, are combined into a new feature vector of length 20. The corresponding EER is 31.32%, which is on the same level as the results of the single statistical features. Hence, these features are not able to describe the subject-specific gait adequately and are not considered anymore in the following evaluation.

Because using all axes performed best for normal walk (setting 1 and 3), all different combinations of the features based on cepstral coefficients using all axes were evaluated. For setting 2 a similar evaluation has been performed using only the x-axis and the magnitude vector. The results for all three settings are given in Table 7.5. One can see that the EERs of setting 1 and 3 vary greatly, although both settings were collected on the same day from the same subjects which are in both cases instructed to walk normally. The difference is, that setting 1 was collected right at the beginning, and the subjects had to get used to the experimental setup. This seems to have a high impact on the stability of their walk, resulting in much higher error rates than obtained with setting 3.

For normal walk (setting 1 and 3) feature set BFCC2MFCC_xyzm is selected as most suitable, because it performs well for both settings. BFCC1BFCC2MFCC_xyzm is the best combination for setting 2, but using only BFCC1_xm still yields a slightly lower EER. In addition, this feature vector is preferred due to its short length.

So far, all given results are cross-day results, meaning that training and testing data are collected on two different days. This is a realistic scenario, as one would not train the models each day. To see the influence of the variation of gait over time, the same-day results were computed, using

Feature Vector	Axes	Length	Setting 1	Setting 3	Axes	Length	Setting 2
BFCC1MFCC	x,y,z,m	104	18.94%	14.14%	x,m	52	14.47%
BFCC2MFCC	x,y,z,m	104	17.28%	14.20%	x,m	52	15.37%
BFCC1BFCC2	x,y,z,m	104	19.13%	13.90%	x,m	52	14.41%
BFCC1BFCC2MFCC	x,y,z,m	156	18.76%	13.86%	x,m	78	14.40%
MFCC	x,y,z,m	52	16.96%	14.94%	x,m	26	16.50%
BFCC1	x,y,z,m	52	19.85%	14.99%	x,m	26	14.37%

Table 7.5.: Cross-day results for different combinations of BFCCs and MFCCs and the best single feature obtained for the three different settings (sampling rate 50 Hz, segment length 5 s).

setting 1 for training and setting 3 for testing. An EER of 5.66% was obtained for feature set BFCC2MFCC_xyzm. This large difference shows the importance of using a testing database containing data of different days.

Varying the Amount of Training Data

So far, only the first twelve normal or fast walks were used for training. In this evaluation the influence of the amount of training data is analyzed for all three settings. For the evaluation using setting 1 and 3, ten different random sequences of the twelve walks were created, which are given in Table B.4 in Appendix B. In addition, the original order of the walks is used. Starting with only one walk for training, successively the other walks are added until the training set contains all walks of the first day of the respective setting. Figures 7.2 and 7.3 show the results for setting 1 and 3. The EERs are given in Table B.6 and B.7 in Appendix B. Comparing the results of the two different settings shows that for setting 1, the order of the training walks has a much higher influence than for setting 3. The reason for this will be the already identified lower stability of the walking of the subjects at the beginning of the data collection. Because the walk of setting 3 is more stable, changing the order of the training walks has less influence.

The median of the eleven different sequences is used for evaluation. For both settings, the general trend is a decrease of the error rates when adding more training data. For setting 1 the lowest median EER of 17.17% is obtained using eleven training walks, all twelve training walks yield an EER of 17.28%. For setting 3 a training set of all twelve walks yields the lowest error rate of 14.20%, using ten walks a similar EER of 14.34% is obtained. Using more than ten walks for training has only minor influence. Using all 24 walks of setting 1 and 3 together for training does decrease the EER to 15.10% when setting 1 is used for testing. This decrease is probably due to the fact that data of setting 3 are of higher quality and not because of the increase of training data. When setting 3 is used for testing the EER is increased to 18.76%. Using the models trained with setting 3 and testing with setting 1 (second day) an EER of 15.99% is obtained for feature set BFCC2MFCC_xyzm. This shows that the models trained with data of setting 3 have better classification abilities.

For evaluation of the influence of the amount of training data for fast walk (setting 2), 16 training walks from the first session are available. As before, the training set is extended by adding additional walks until all 16 walks are used for training. The used random sequences are given in Table B.5 in Appendix B. The results for BFCC1_xm are depicted in Figure 7.4; the EERs are given in Table B.8 in Appendix B. The median is used for evaluation. When increasing the amount

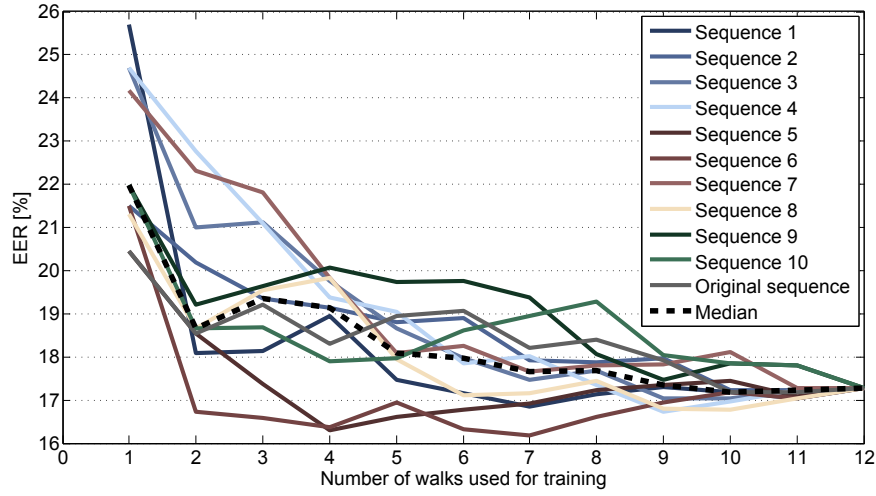


Figure 7.2.: EERs obtained when varying the amount of training data from 1 to 12 walks. Results for ten different random training sequences and the original sequence are given for setting 1 (normal walk), sampling rate 50 Hz, segment length 5 s. In addition, the median of the eleven results is plotted.

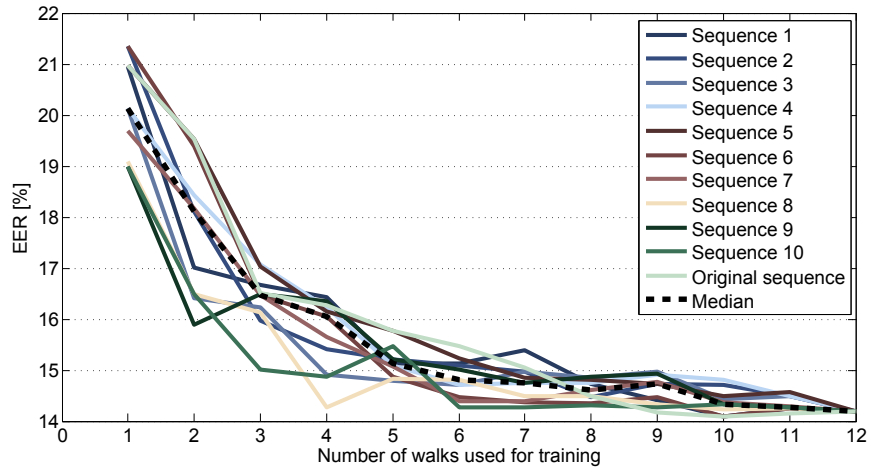


Figure 7.3.: EERs obtained when varying the amount of training data from 1 to 12 walks. Results for ten different random training sequences and the original sequence are given for setting 3 (normal walk), sampling rate 50 Hz, segment length 5 s). In addition the median of the eleven results is plotted.

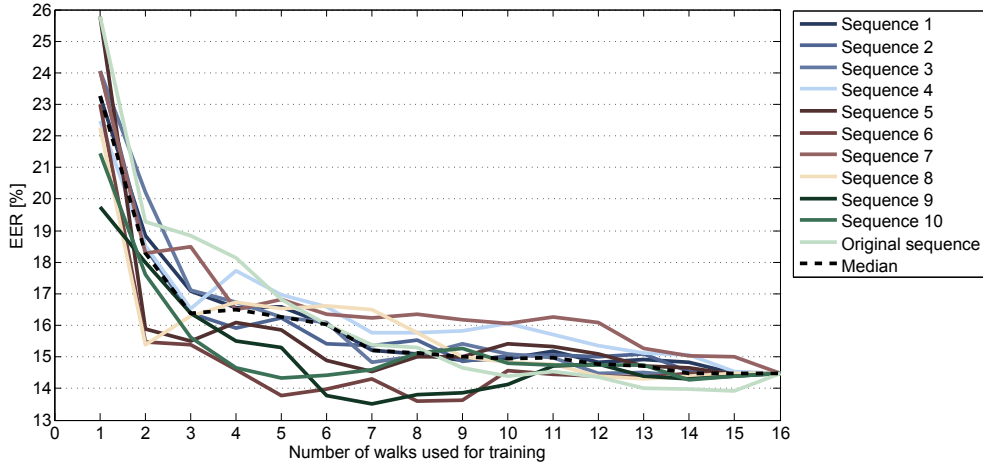


Figure 7.4.: EERs obtained when varying the amount of training data from 1 to 16 walks for feature set BFCC1_xm. Results for ten different random training sequences and the original sequence are given for setting 2 (fast walk), sampling rate 50 Hz, segment length 5 s. In addition, the median of the eleven results is plotted.

of training data, a high decrease of the EER from 23.26% to 15.20% is visible up to a training set consisting of seven walks. Adding more walks does only decrease the EER slightly. Lowest EER is 14.47% when using 15 training walks. Using all 16 walks results in an EER of 14.48%.

For all three settings, training data consisting of 4.5 minutes is sufficient to train the HMMs. This corresponds to ten normal walks or 14 fast walks (see Table 7.2).

Mixed Velocities

In the previous evaluations, training and testing data were always belonging to the same walking velocity. For setting 1 and 3 the subjects were told to walk in normal speed, for setting 2 they should walk fast. Now, a cross-setting evaluation is performed, using data from one velocity for training and the other velocity for testing. The results are given in Table 7.6. Feature set is BFCC2MFCC_xyzm and the first twelve walks of the first session are used for training, the first twelve walks of the second session are used for testing. In addition, mixed walks (fast and normal) are used for training (six walks from each setting) and each of the settings is separately used for testing.

The cross-velocity results are marked gray in the table. It is clearly visible that the obtained EERs are much worse than the ones obtained for the same-velocity comparisons. The results obtained from models which have been trained with data of both velocities (S1+S2, S2+S3) show only a small degradation of the EER. Therefore, one can conclude that the best results are obtained when the training and testing data are collected when walking approximately in the same velocity. Changing the speed significantly, e.g. by walking fast although the models have been trained with normal walk, does highly degrade the biometric performance. A suitable solution is to train the models with data of different speeds, which only increases the error rates slightly.

Train Setting	Test Setting		
	S1	S2	S3
S1	17.28	40.09	19.85
S2	42.51	17.20	38.84
S3	15.99	37.40	14.20
S1+S2	19.12	23.39	21.06
S1+S3	14.42	38.27	17.46
S2+S3	22.01	21.49	18.18

Table 7.6.: EERs when using different combinations of training and test settings. Cross-velocity results are marked gray. Feature set is BFCC2MFCC_xyzm for all settings (sampling rate 50 Hz, segment length 5 s).

Voting Results

For each setting it is analyzed if applying the voting algorithm further decreases the EER. The classifier is trained using all twelve walks of the first session. Probe data are taken from the first twelve walks of the second session. The number of votes ($\#V$) was varied from 10 to the maximal possible number 60 and the number of required votes for genuine $\#GV$ from 1 to 5.

For normal walk, this evaluation is performed using feature set BFCC2MFCC_xyzm and the results are given in Figure 7.5 and 7.6. Using $\#V = 10$ is not sufficient, using $\#V = 30$ gives good results for both settings. The influence of the number of required genuine votes differs for both settings but $\#GV = 4$ gives good results for both. Therefore, the combination of $\#V = 30$ and $\#GV = 4$ is considered to be most suitable, reducing the EER from 17.28% to 13.89% for setting 1 and from 14.20% to 8.47% for setting 3. The thirty votes correspond to around 75 seconds of walking.

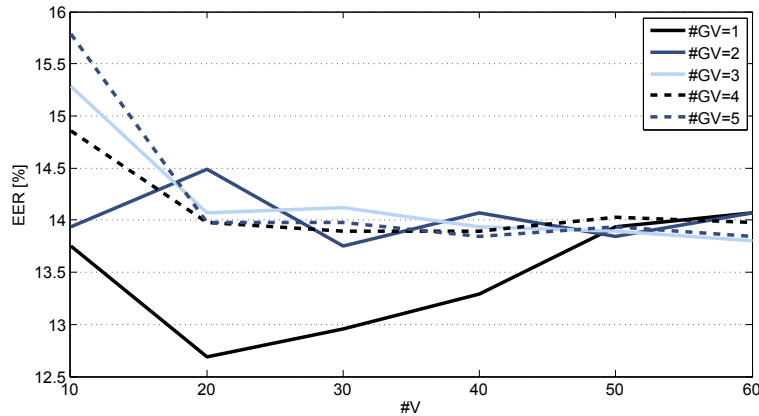


Figure 7.5.: Voting results for setting 1 using different numbers of votes ($\#V$) and genuine votes ($\#GV$).

Results for setting 2 are given in Figure 7.7. The maximal available number of segments to group has been 60. The number of required votes for genuine $\#GV$ has no big influence to the

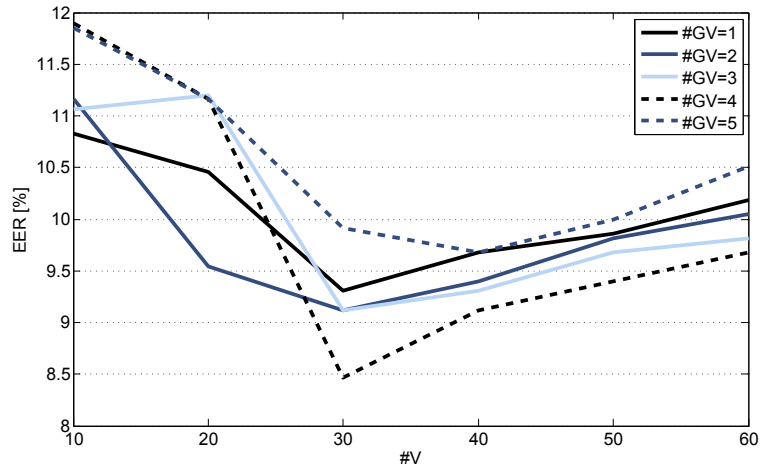


Figure 7.6.: Voting results for setting 3 using different numbers of votes ($\#V$) and genuine votes ($\#GV$).

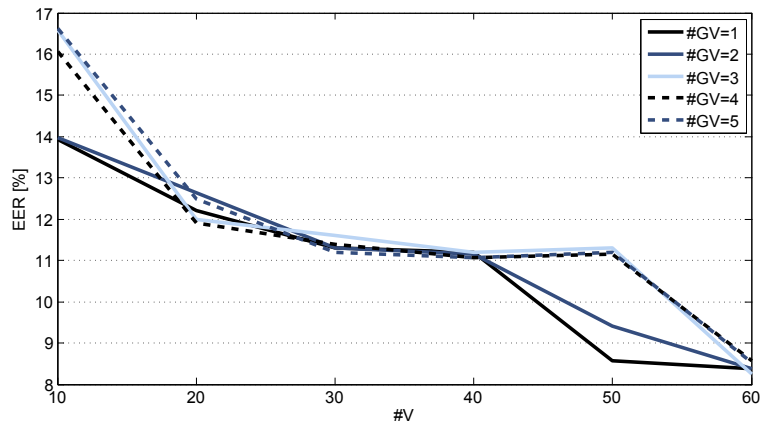


Figure 7.7.: Voting results for setting 2 using different numbers of votes ($\#V$) and genuine votes ($\#GV$).

results, a larger group size $\#V$ is decreasing the error rates. Best result is obtained for $\#V = 60$ (around 2.5 minutes of walking) and $\#GV = 3$. This decreases the EER from 14.48% to 8.24%.

7.3. Support Vector Machines

This evaluation of SVMs follows a similar procedure as the evaluation of HMMs, to allow for a straight forward comparison of the approaches. Sampling rates 25, 50, 100, and 200 Hz were analyzed for segment lengths between 2 and 7.5 s, which again were overlapping by 50%. From each segment the features presented in Section 5.3 are extracted. The results are partly published in [105], where they are compared to the HMM results.

7.3.1. Evaluation

Single and combined features as well as the influence of sampling rate and segment length are analyzed. The amount of training data is varied to identify the optimal training duration. The algorithm does only return the classification result for each probe segment, no scores are given. Therefore, it is only possible to compute FMR and FNMR, the EER cannot be determined. For a better comparison to the HMM results the HTER is given, which is half of the sum of FMR and FNMR. At the end, the voting algorithm is applied to get the final error rates. Varying the input parameters $\#GV$ and $\#V$ yields various combinations of FMR and FNMR and allows in some cases to determine the EER (only if a combination of $\#GV$ and $\#V$ produces an FMR which is higher than the corresponding FNMR).

Single Features

The single features were analyzed separately, using the first twelve normal walks from the first session for training and testing with the first twelve normal walks of the second session. Results for the evaluation of single features are given in Table 7.7. Similar to the HMM results, BFCCs and MFCCs outperform the statistical features. Best single feature is BFCC2_xyzm. To validate the correct choice of the Bark scale, it is replaced by a linear scale as described in Section 5.3.1. This does slightly increase the HTER from 24.08 to 24.52. Hence, the influence is only small, but the Bark scale outperforms the linear scale.

Because they remain the best features, only the results for BFCC and MFCC for different axis combinations are stated in Table 7.8 for setting 2 (fast walk). The SVMs were trained using all 16 walks of the first session and the twelve first fast walks of the second session were used to calculate the probe data.

Combined Features

The influence of different sampling rates and segment lengths for feature set BFCC1BFCC2_xm is shown in Table 7.9. FMR and FNMR for setting 1 for all evaluated feature sets are given in Appendix C. For setting 1 the best HTER of 18.87 is obtained using sampling rate 50 Hz and a segment length of 6 s. These preprocessing configurations will be used in the following evaluations for normal walk settings (1 and 3). For fast walk (setting 2) the combination of sampling rate 50 Hz and segment length 7.5 s yielded best results and is used for evaluations of fast walk data.

To validate that using only x-acceleration and the magnitude vector is the best choice, the feature vector BFCC1BFCC2 is computed for each axis separately. In addition, one feature vector

Feature Type	Feature Length	FMR	FNMR	HTER
Max_xyzm	4	4.53	84.27	44.40
Min_xyzm	4	3.05	80.47	41.76
Mean_xyzm	4	0.61	93.46	47.03
Std_xyzm	4	4.43	72.06	38.25
RMS_xyzm	4	4.17	75.20	39.69
Cros_xyzm	4	1.06	92.44	46.75
Diff_xyzm	4	0.10	99.71	49.90
Bin5_xyzm	20	3.12	69.59	36.36
Bin10_xyzm	40	2.97	63.87	33.42
BFCC1_xyzm	52	0.45	48.46	24.45
BFCC2_xyzm	52	0.57	47.59	24.08
MFCC_xyzm	52	0.37	50.70	25.53

Table 7.7.: Results for single features when using sampling rate 50 Hz and segment length 6 s.

	BFCC1	BFCC2	MFCC
x	21.17	22.95	23.64
xm	17.91	13.75	19.76
xyzm	22.65	21.63	23.14

Table 7.8.: Fast walk results for single features and different axes (HTER) using sampling rate 50 Hz, segment length 7.5 s, and 16 walks for training.

	Setting 1					Setting 2			
	25 Hz	50 Hz	100 Hz	200 Hz		25 Hz	50 Hz	100 Hz	200 Hz
2 s	22.86	21.73	22.25	22.78	2 s	21.91	19.66	19.67	19.81
3 s	22.41	23.96	24.44	24.57	3 s	19.19	17.56	17.96	17.83
4 s	20.61	23.04	19.54	19.53	4 s	18.56	17.25	18.29	17.59
5 s	20.73	20.52	20.44	20.61	5 s	16.95	16.89	17.09	17.29
6 s	20.57	18.87	18.92	19.30	6 s	16.37	17.09	17.40	17.45
7.5 s	20.09	20.06	20.34	20.26	7.5 s	15.67	13.75	13.77	14.03

Table 7.9.: HTER for feature set BFCC1BFCC2_xm for normal and BFCC2_xm for fast walk at various preprocessing configurations using 12 normal or 16 fast walks for training, respectively.

was created for each axis separately, consisting of all previously evaluated features. The results are given in Table 7.10. For both kinds of feature vectors, using only the x-axis or concatenating x-axis and magnitude vector gives better results than the other axes. Feature vector BFCC1BFCC2_xm yields the best HTER of 18.87 and has the advantage of being much shorter than the vector containing all features. The evaluation using testing and training data from the same day results in a much lower HTER of 6.94 for the BFCC1BFCC2_xm feature set. Therefore, the following evaluation will continue using cross-day comparisons.

Axis	allFeatures				BFCC1BFCC2			
	Feature Length	FMR	FNMR	HTER	Feature Length	FMR	FNMR	HTER
x	61	1.89	39.80	20.85	13	1.77	42.82	22.92
y	61	2.06	68.31	35.19	13	1.91	69.83	35.87
z	61	1.37	65.23	33.30	13	1.62	69.74	35.68
m	61	1.50	49.80	25.65	13	1.74	53.14	27.44
xm	122	0.40	41.51	20.95	26	1.20	36.54	18.87
xyzm	244	0.01	66.83	33.42	52	0.22	52.82	26.52

Table 7.10.: Error rates obtained separately for the different axes (sampling rate 50 Hz, segment length 6 s).

The results for the identified best axes and different combinations of cepstral coefficients are given in Table 7.11. For setting 1 and 3 the tests showed that the feature vector containing BFCC1 and BFCC2 from x-axis and magnitude vector (BFCC1BFCC2_xm) performs best, yielding an HTER of 18.87 and 17.59, respectively. For setting 2 feature set BFCC1BFCC2_xm is the best combination, as well, but the single feature BFCC2_xm remains best.

Feature Vector	Feature Length	Setting 1		Setting 2		Setting 3	
		FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1BFCC2_xm	52	1.20%	36.54%	1.09%	26.54%	1.23%	33.95%
BFCC1MFCC_xm	52	1.49%	38.14%	1.14%	29.41%	0.25%	63.28%
BFCC2MFCC_xm	52	1.12%	39.68%	1.31%	30.68%	0.99%	36.76%
BFCC1BFCC2MFCC_xm	78	0.96%	40.03%	0.80%	30.97%	0.82%	39.16%

Table 7.11.: Cross-day results for different combinations of BFCC and MFCC for x-axis and magnitude vector combined, for setting 1, 2, and 3. For each setting the maximal amount of training data and the best preprocessing configuration is used.

Varying the Amount of Training Data

The results stated so far were achieved using the first 12 normal or 16 fast walks for training. Now, the amount of training data is varied from using only one walk up to using all available walks as before. The same ten random sequences are used as in the HMM-evaluation. Figure 7.8 and 7.9 show the results for setting 1 and 3.

Figure 7.10 gives the results for setting 2 using the ten random sequences presented in Table B.5. The general trend visible for all three settings is a lowering of the error rates when the amount of training data is increased. But there are some outliers. For example, training SVMs with walks

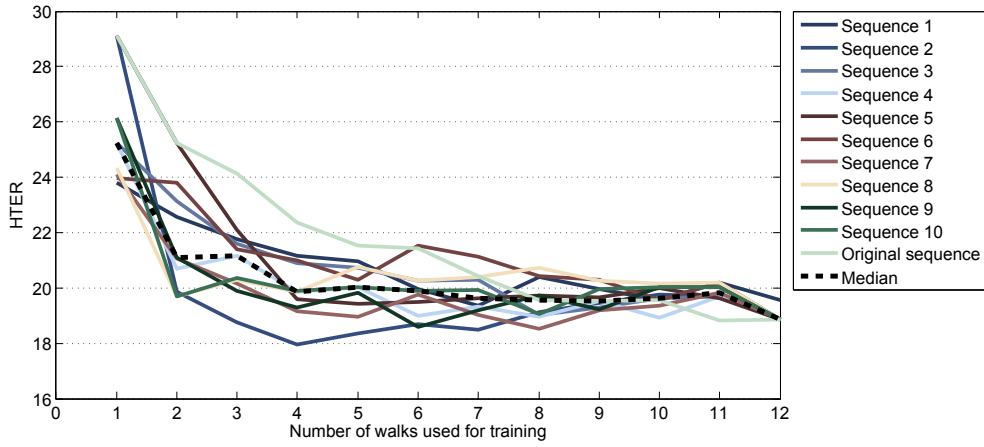


Figure 7.8.: Results for SVMs obtained when varying the amount of training data from 1 to 12 walks. Results for ten different random training sequences and the original sequence are given for setting 1 (normal walk), sampling rate 50 Hz, segment length 6 s, and feature vector BFCC1BFCC2_xm. In addition, the median of the eleven results is plotted.

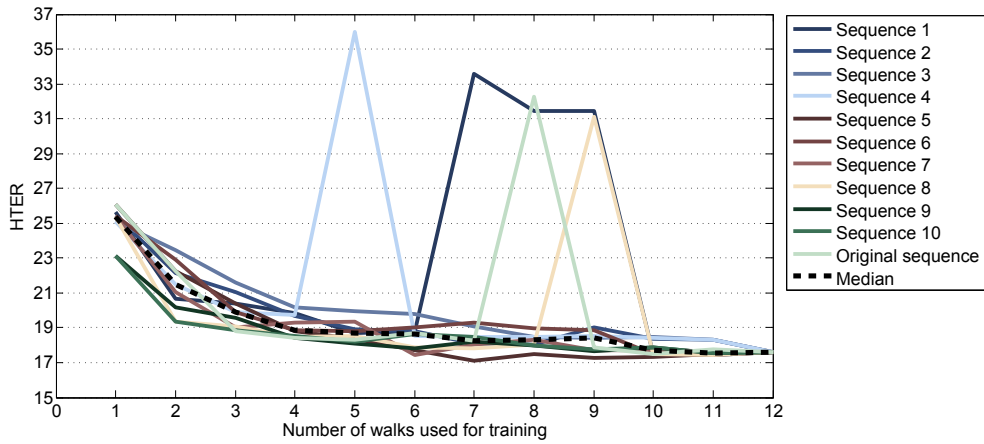


Figure 7.9.: Results for SVMs obtained when varying the amount of training data from 1 to 12 walks. Results for ten different random training sequences and the original sequence are given for setting 3 (normal walk), sampling rate 50 Hz, segment length 6 s, and feature vector BFCC1BFCC2_xm. In addition, the median of the eleven results is plotted.

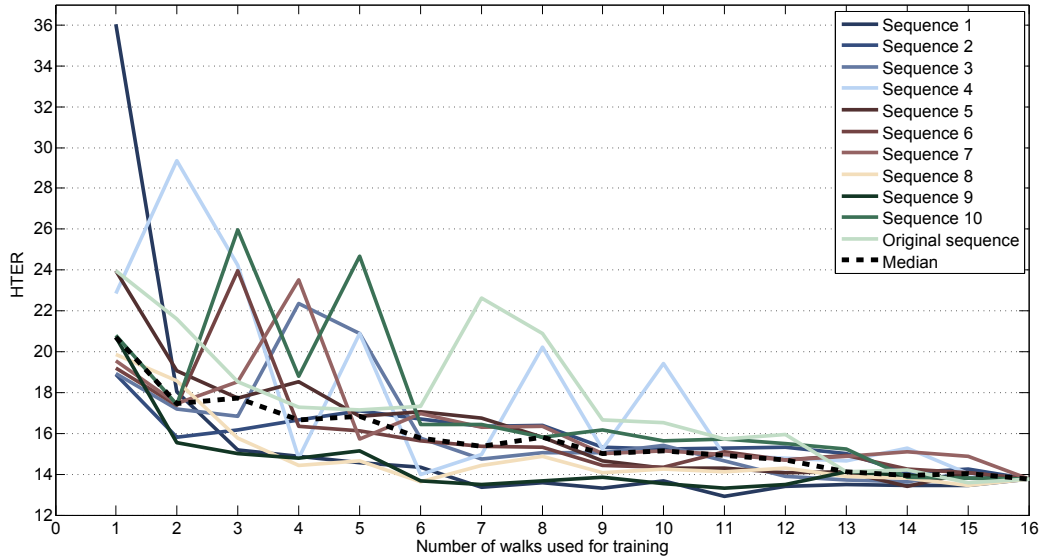


Figure 7.10.: Results for SVMs obtained when varying the amount of training data from 1 to 16 walks. Results for ten different random training sequences and the original sequence are given for setting 2 (fast walk), sampling rate 50 Hz, segment length 7.5 Hz, and feature vector BFCC2.xm. In addition, the median of the eleven results is plotted.

1 to 8 of setting 3 (original sequence) results in very high error rates, indicating that the last added walk had major impact on the position of the support vectors. Adding a further walk does improve the situation again. This property of the SVM-algorithm is a disadvantage compared to the previously evaluated HMMs where a smooth decrease of the EER could be observed. As before, varying the order of the training walks had more impact for setting 1 than for setting 3, which again yields lower HTERs. Using two minutes (four walks) for training is sufficient in case of normal walk. This decreases the median HTER from 25.25 to 19.85 for setting 1 and from 25.36 to 19.08 for setting 3. Using the maximal amount of twelve walks does further decrease the HTER to 18.87 and 17.59 for the two different settings, respectively.

The training sequences for setting 2 yield more outliers than setting 3. In case of fast walk, using 13 walks (corresponding to 4.5 minutes of data) gives acceptable results. This decreases the median HTER from 20.71 to 14.13. Using all available 16 walks yields an HTER of 13.75.

Mixed Velocities

The evaluation in Section 7.2.1 showed that there is a high degradation of the error rates in case reference and probe data are not collected from the same walking velocity. Now, it is analyzed if a corresponding behavior can be observed for SVMs. Table 7.12 shows the results. The cross-velocity error rates are marked gray. The SVMs were trained using twelve walks of the first session. In case two settings were used for training, six walks were taken from each. Test data consisted of twelve walks of the second session. For all settings, feature set BFCC1BFCC2.xm is used. The cross-velocity results, obtained when using setting 1 and/or 3 for training and using setting 2 for

testing or vice versa, are much worse than the same-velocity results. As for HMMs, using data of both velocities for training gives acceptable results for probe data of both velocities.

Train Setting	Test Setting		
	S1	S2	S3
S1	18.87	46.72	25.49
S2	46.79	14.66	43.96
S3	20.05	43.87	17.59
S1+S2	18.59	18.65	26.71
S1+S3	14.97	43.18	18.04
S2+S3	22.23	18.27	18.84

Table 7.12.: HTERs when using different combinations of training and test settings. Cross-velocity results are marked gray. Feature set is BFCC1BFCC2.xm for all settings (sampling rate 50 Hz, segment length 6 s).

Voting

The results stated so far for SVMs are highly unbalanced between FMR and FNMR. While the FMR is very low, the FNMR is still on an unacceptable level. To improve this situation, the voting algorithm is applied. For setting 1 and 3, SVMs are trained using all 12 normal walks of the first session of the respective setting, and 12 normal walks of the second session are used for testing. The number of votes ($\#V$) is varied from 10 to the maximal possible number 60 and the number of required votes for genuine $\#GV$ from 1 to 10. The results for feature set BFCC1BFCC2.xm are given in Figure 7.11 for setting 1. The resulting HTERs are between 8.68 and 27.96. Using $\#GV=1$ yields best results independently of the number of votes considered. The best result was obtained when using $\#V = 50$ and $\#GV = 1$. At that point the FMR is 6.25% and the FNMR equals 11.11%.

Figure 7.12 gives the results for setting 3. As before, requiring only one vote for genuine ($\#GV = 1$) gives best results and there is not a big influence of the group size $\#V$. The best result of FMR = 5.18% and FNMR = 16.67% is obtained for $\#V = 40$ and $\#GV = 1$. These 40 votes are considered to be a suitable group size for normal walk, because they give good results for both settings and keep the amount of necessary test data low. Because one segment is of length 6 seconds and the segments overlap by 50%, the 40 votes correspond to around 2 minutes of walking.

Voting results for setting 2 (fast walk) using feature set BFCC2.xm are given in Figure 7.13. A different behavior than for the other two settings is visible. The best result is obtained using the maximal group size of $\#V = 50$ and a high amount of at least $\#GV = 7$. The corresponding result is an HTER of 6.61 (FMR = 3.22% and FNMR = 10.00), the required walking duration is around 3.5 minutes. The results for setting 2 allow for a computation of the EER, which is 8.29%. The closest combination of FMR = 7.50% and FNMR = 8.33% is achieved for $\#V = 35$ and $\#GV = 1$.

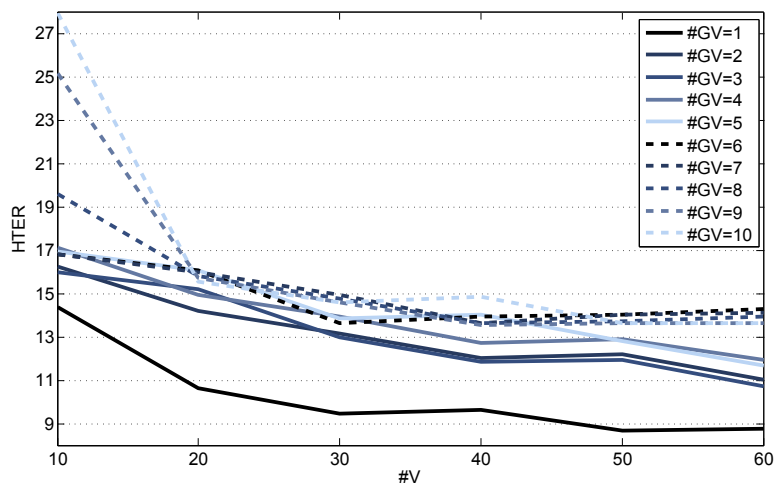


Figure 7.11.: Voting results for SVMs and setting 1. Twelve walks are used for training; data are preprocessed using sampling rate 50 Hz and segment length 6 s.

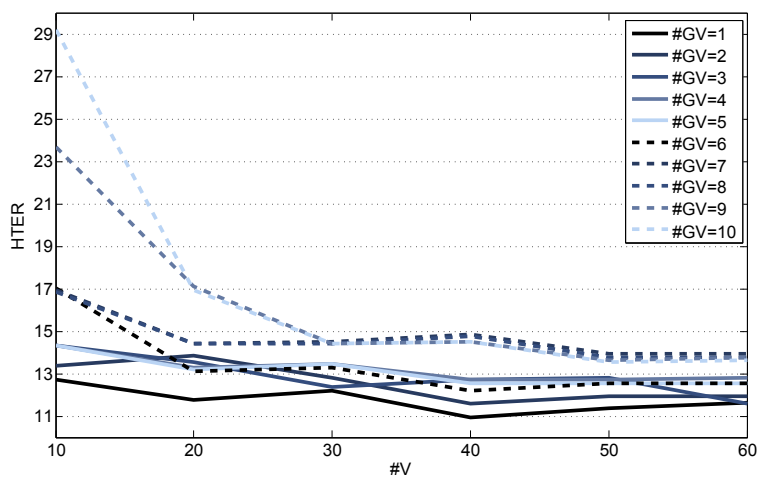


Figure 7.12.: Voting results for SVMs and setting 3. Twelve walks are used for training; data are preprocessed using sampling rate 50 Hz and segment length 6 s.

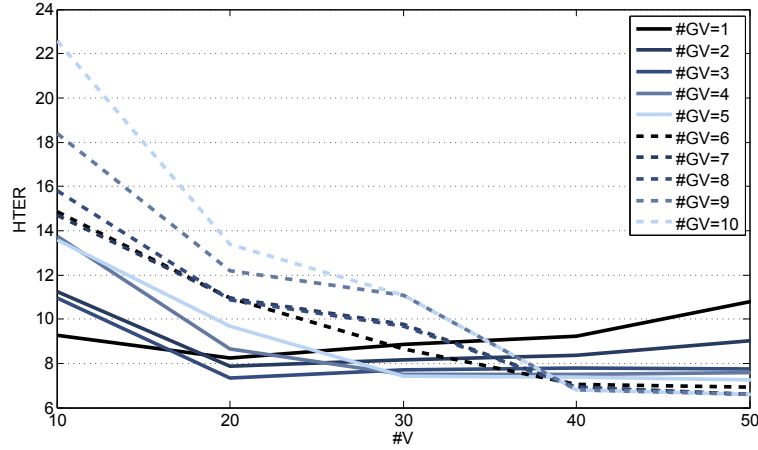


Figure 7.13.: Voting results for SVMs and setting 2. 16 walks are used for training; data are preprocessed using sampling rate 50 Hz and segment length 7.5 s.

7.4. *k*-Nearest Neighbor Algorithm

Because of the promising results obtained using HMMs and SVMs, further machine learning algorithms² were evaluated concerning their computational and biometric performance. The *k*-NN algorithm outperformed the other algorithms in this initial test and was chosen for further evaluation. The results are published in [111].

7.4.1. Evaluation

The evaluation is kept close to the process used previously. Statistical features are not considered anymore as single features, instead the DPS-features introduced in Section 5.3.2 are evaluated. As before, the influence of the amount of training data and different walk velocities is analyzed and the voting algorithm is applied to reduce the FNMR.

Single Features

Table 7.13 shows the HTER for MFCC and BFCC computed for different axes-combinations and obtained using different values for the number of considered neighbors (*k*). The chosen sampling rate is 100 Hz and the segments are of length 7.5 s. There is hardly any influence of the number of considered neighbors, therefore it is set to 3. The combination of all three axes and the magnitude vector gives best results for all three features when normal walk data (from setting 1 or 3) are used. Best feature set is BFCC2_xyzm obtaining an HTER of 13.09 for setting 1. When fast walk data are used (setting 2, using all 16 walks for training) the combination of x-axis and magnitude vector for BFCC1 gives best results.

The influence of the sampling rate and segment length is shown in Table 7.14 for setting 1. The influence of both parameters is low. A sampling rate of 100 samples per second and a segment length of 7.5 s is used for the following evaluations of the *k*-NN algorithm. Again, it is interesting

²All algorithms of the WEKA-library for which an Android port exists (e.g. random forest, multilayer perceptron).

to see if a linear scale instead of the Bark scale would improve the so far best results of feature set BFCC2_xyzm. Using the same configuration but replacing the Bark scale by the linear scale increases the HTER from 13.09 to 14.28, confirming the correct choice of the Bark scale.

k	Setting 1				Setting 2	Setting 3
	1	2	3	4	3	3
BFCC1_xyzm	17.48	16.83	17.56	17.17	15.13	18.74
BFCC2_xyzm	13.46	12.62	13.09	12.61	14.38	12.25
MFCC_xyzm	15.14	14.27	14.98	14.23	14.88	12.34
BFCC1_xm	18.36	16.70	18.39	17.29	11.83	14.84
BFCC2_xm	14.63	14.09	14.85	14.47	13.26	13.24
MFCC_xm	16.79	15.61	16.53	16.02	13.25	15.51
BFCC1_x	23.13	20.76	22.74	21.54	21.11	24.73
BFCC2_x	22.35	20.48	22.15	21.06	22.11	18.80
MFCC_x	24.58	21.98	24.59	23.00	23.26	20.31

Table 7.13.: HTER for BFCC and MFCC using different values for the number of considered neighbors k (sampling rate is 100 Hz and the segments are of length 7.5 s).

	3 s			5 s			7.5 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
50 Hz	3.78	26.75	15.26	3.67	24.72	14.20	3.96	23.59	13.77
100 Hz	3.95	27.15	15.55	3.78	24.46	14.12	3.97	22.22	13.09
200 Hz	4.15	27.02	15.59	3.88	24.47	14.17	4.11	22.26	13.18

Table 7.14.: Error rates for feature set BFCC2_xyzm for different sampling rates and segment lengths (setting 1).

Combined Features

The statistical features have not been considered as single features in this evaluation but they are included in the DPS feature sets which are evaluated now. Table 7.15 gives the results for the DPS feature vectors, which are defined in Appendix H. The best results using the DPS features are an FMR of 4.24% at an FNMR of 23.44%, which is obtained by using 0.1 as a threshold to create the feature set. The corresponding feature vector DPS0.1 has 186 entries, hence only 58 features are not considered. The deleted features are mainly statistical ones, like *Diff*, but also entries at the end of the cepstral coefficient vectors are in some cases removed. The result of DPS0.1 is similar to the so far obtained best result of an FMR of 3.97% at an FNMR of 22.22%. Because feature vector DPS0.1 is much longer and its creation requires the calculation of several different features, using only BFCC2_xyzm is preferred.

Table 7.16 shows the results obtained for different combinations of BFCC1, BFCC2, and MFCC, using the maximal amount of training data available for each setting. Only the best axes-combinations are presented for each setting. This has been the combination of all three axes and the magnitude vector for setting 1, and the combination of x-axis and magnitude vector for

	Threshold	Feature Length	FMR	FNMR	HTER
DPS-0.4	-0.4	241	4.54	25.42	14.98
DPS-0.3	-0.3	240	4.55	25.38	14.97
DPS-0.2	-0.2	238	4.58	25.30	14.94
DPS-0.1	-0.1	230	4.55	24.66	14.60
DPS0	0	217	4.43	23.51	13.97
DPS0.1	0.1	186	4.24	23.44	13.84
DPS0.2	0.2	160	4.18	23.67	13.93
DPS0.3	0.3	119	4.18	27.97	16.08
DPS0.4	0.4	77	4.49	33.57	19.03
DPS0.5	0.5	44	4.34	40.09	22.22
DPS0.6	0.6	24	4.28	49.81	27.05
DPS0.7	0.7	12	5.48	61.51	33.50

Table 7.15.: Error rates obtained by using feature sets consisting of all features with a DPS above the given threshold (sampling rate 100 Hz, segment length 7.5 s).

setting 2 (see Table 7.13). The combinations yield slightly worse results than the single features BFCC2_xyzm and BFCC1_xm for setting 1 and 2, respectively.

Features	Setting 1				Setting 2			
	Axes	FMR	FNMR	HTER	Axes	FMR	FNMR	HTER
BFCC1MFCC	x,y,z,m	4.09	26.52	15.31	x,m	4.10	22.49	13.29
BFCC2MFCC	x,y,z,m	3.90	24.28	14.09	x,m	4.26	22.53	13.39
BFCC1BFCC2	x,y,z,m	4.24	24.77	14.51	x,m	4.44	21.83	13.14
BFCC1BFCC2MFCC	x,y,z,m	4.01	23.63	13.82	x,m	4.32	21.83	13.08

Table 7.16.: Error rates for different combinations of BFCCs and MFCCs for setting 1 and 2 (sampling rate 100 Hz, segment length 7.5 s).

Varying the Amount of Training Data

For the two normal walk settings, Figures 7.14 and 7.15 state the results for varying amounts of training data obtained using the same ten random sequences as before. The results are given for feature set BFCC2_xyzm. Analyzing the median HTER obtained, increasing the number of walks for training has the same impact for both settings. Adding more walks significantly decreases the HTER up to using a set of four walks. For setting 1 and feature vector BFCC2_xyzm the initial HTER of 16.96 is lowered to 13.61. Using the same amount of training data for setting 3 decreases the HTER from 15.59 to 12.71. Afterwards, the decrease is slower. For both settings, the lowest error rate of 13.05 or 12.06 is obtained when using eleven or ten walks for training, respectively.

For setting 2 the progress for feature set BFCC1_xm is depicted in Figure 7.16. At least nine walks should be used for training, reducing the HTER from 17.36 to 12.02. The minimal HTER of 11.81, considering the median again, is obtained when using 15 walks for training.

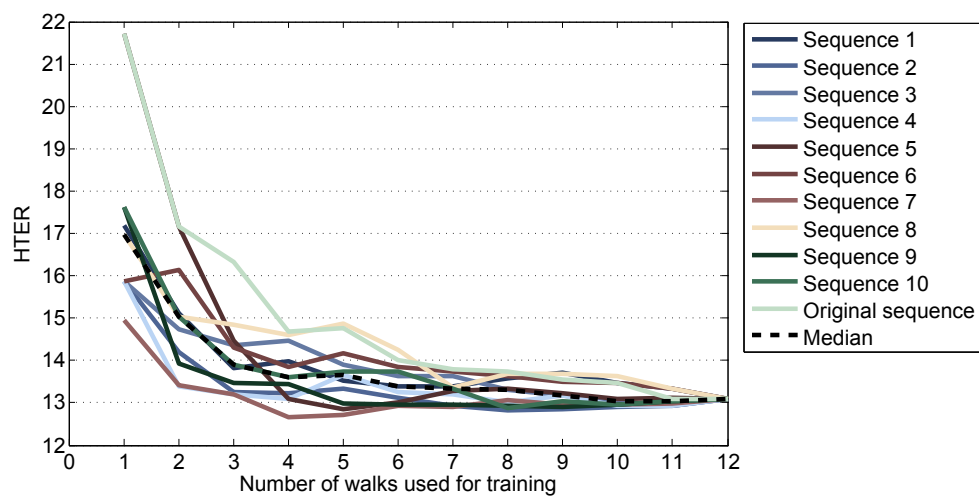


Figure 7.14.: Influence of the amount of training data for setting 1 when using feature set BFCC2_xyzm (sampling rate 100 Hz, segment length 7.5 s).

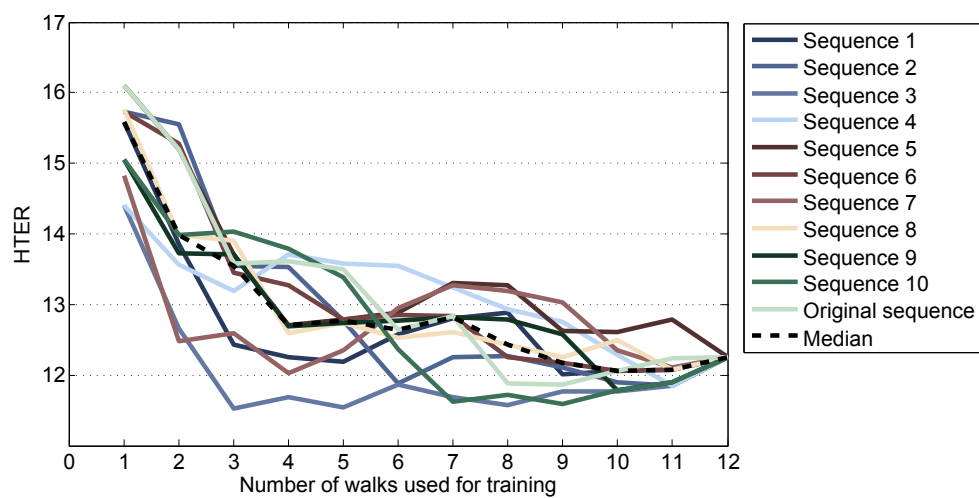


Figure 7.15.: Influence of the amount of training data for setting 3 when using feature set BFCC2_xyzm (sampling rate 100 Hz, segment length 7.5 s).

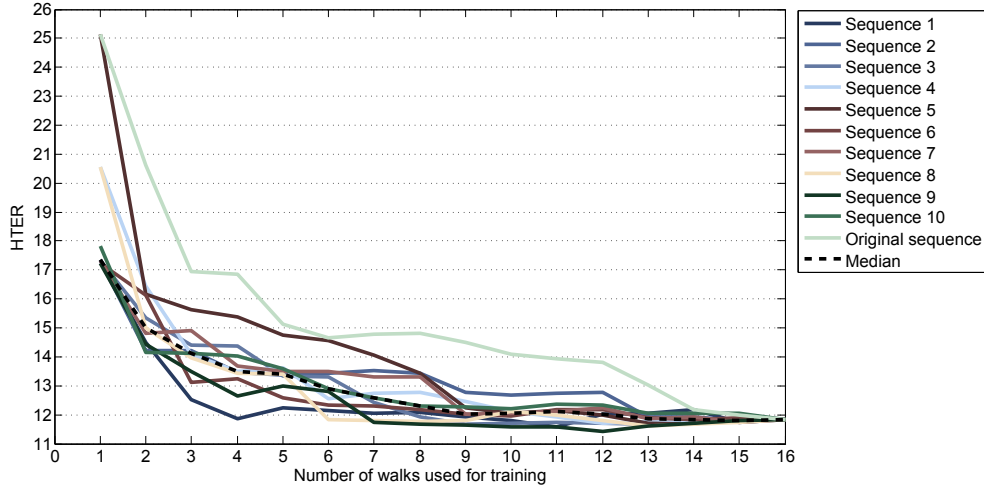


Figure 7.16.: Influence of the amount of training data for setting 2 when using feature set BFCC1_xm (sampling rate 100 Hz, segment length 7.5 s).

	Test Setting		
	S1	S2	S3
S1	13.09	41.57	17.28
S2	45.26	14.79	42.42
S3	14.97	38.37	12.25
S1+S2	15.45	14.85	18.21
S1+S3	12.37	38.24	13.82
S2+S3	15.60	15.16	14.76

Table 7.17.: HTERs when using different combinations of training and test settings. Cross-velocity results are marked gray. Feature set is BFCC2_xyzm for all settings (sampling rate 100 Hz, segment length 7.5 s).

Mixed Velocities

Now it is evaluated if using data of different walking velocities for training and testing does influence the biometric performance of the *k*-NN algorithm as much as for HMMs and SVMs. Table 7.17 gives the results for the different setting combinations of normal walk (setting 1 and 3) and fast walk (setting 2). The cross-velocity results are marked gray. For all settings, training data consisted of twelve walks of the first session, test data of twelve walks of the second session. For training with mixed settings, the training data consisted of six walks of each setting. Feature set BFCC2_xyzm is used for all three settings. As before, the cross-velocity performance is poor, but acceptable error rates can be obtained by using training data of both velocities.

Voting Results

Similar to the results for SVMs, the FNMR is still very high, while the FMR is low. To get a more balanced situation the voting algorithm is applied. Results for setting 1 are given in Figure 7.17,

Figure 7.18 gives the results for setting 3. Aiming at best results for both settings, $\#V = 25$ and $\#V = 4$ is considered as the best setting. This is decreasing the HTER from 13.09 to 8.98 for setting 1 and from 12.25 to 9.91 for setting 3.

For setting 2 the highest number of available votes is $\#V = 39$ because of the shorter duration of the walks due to the higher velocity. Best results are obtained for $\#V = 35$ and $\#V = 3$, decreasing the HTER from 14.79 to 9.17. For all three settings the EER can be computed from the voting results. For setting 1 EER = 9.44%, for setting 2 EER = 9.72%, and for setting 3 EER = 7.64%.

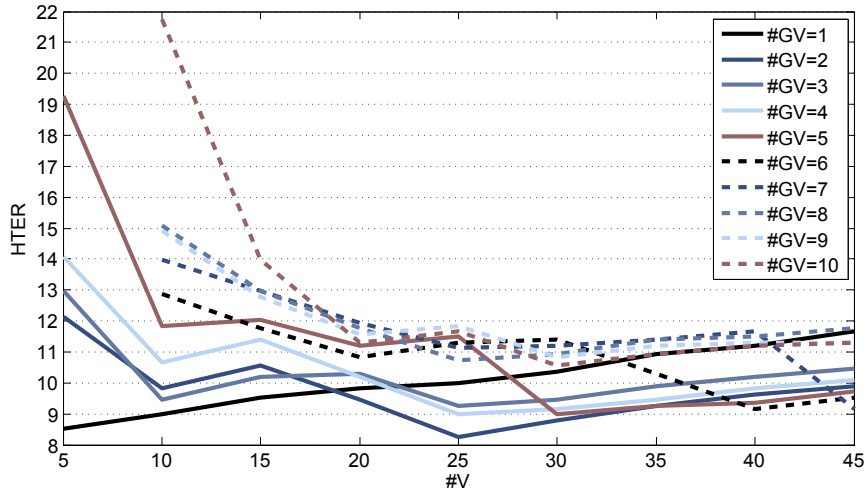


Figure 7.17.: k-NN voting results for setting 1, feature set BFCC2_xyzm (sampling rate 100 Hz, segment length 7.5 s).

7.4.2. Implementation as MBASSy Module

The k-NN algorithm has been implemented as MBASSy module using the previously identified best preprocessing configuration and feature set BFCC2_xyzm. During enrolment the subject is requested to walk for five minutes. Afterwards the classifier is constructed using the enrolment data of the genuine subject and data of 20 subjects from database 2 (setting 1, session 1) as impostor data. Using pre-computed feature vectors from the impostor data, training takes around 1.5 minutes on a Motorola Defy smartphone running Android 2.2. This is a short enough time, because enrolment is not performed often. The effort for the user is restricted to five minutes of walking. To obtain the authentication result, iteratively accelerometer data can be collected and the classification can be performed. Extracting features from 30 seconds of accelerometer data and performing the classification takes around seven seconds. The obtained classification result can be stored in MBASSy and the most current one can be returned without delay whenever an authentication result is required.

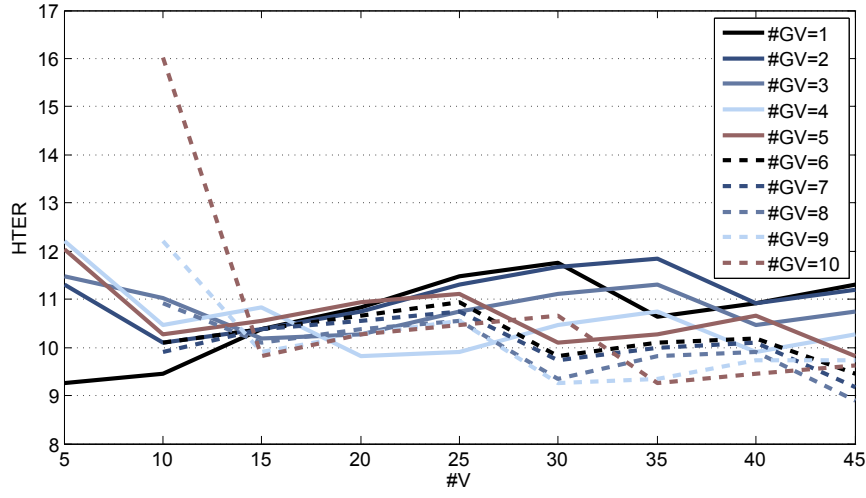


Figure 7.18.: *k*-NN voting results for setting 3, feature set BFCC2_xyzm (sampling rate 100 Hz, segment length 7.5 s).

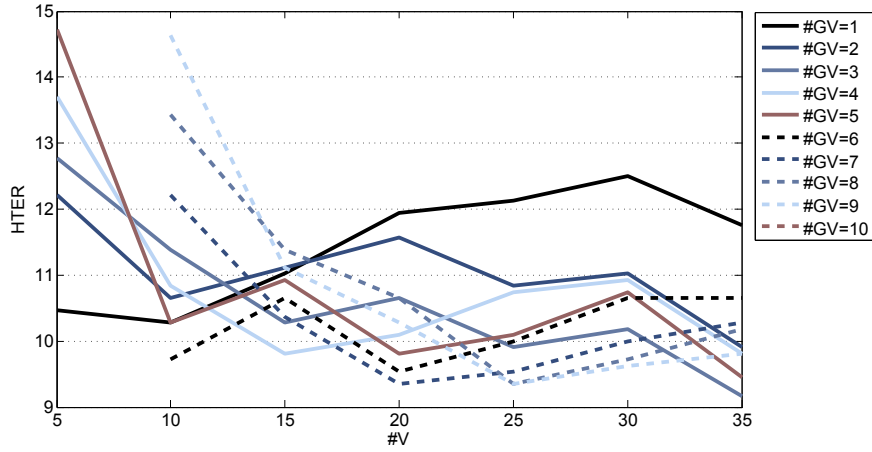


Figure 7.19.: *k*-NN voting results for setting 2, feature set BFCC1_xm (sampling rate 100 Hz, segment length 7.5 s).

7.5. Combining the Cycle-based Segmentation with Classification via HMMs

Normally, either gait cycles containing raw accelerometer data are used as feature vectors and compared using template-based methods or features are extracted from fixed-length time segments and compared using machine learning methods. Now, the two approaches are combined. The results are published in [108]. Using data from setting 1, it is evaluated if a cycle-based segmentation is preferred to a fixed-length time segmentation. In addition, instead of extracting features from the segments, the raw data are used for training of the models. By replacing the fixed-length time segmentation by the cycle extraction method, it is assured that each segment contains the same amount of steps. First, cycles are extracted from the acceleration data. After omitting the unusual cycles, between five and 33 cycles remain per subject for each walk. BFCC and MFCC features are extracted from these cycles, but also the time series data are directly used. HMMs are applied for classification. The results are compared with the previous ones using fixed-length time segments.

7.5.1. Varying the Sampling Rate

To evaluate the influence of the sampling rate, the cycles were normalized to a length of 100 samples during the cycle extraction process. The results for sampling rate 50, 100, and 200 are given as DET-curves in Figure 7.20. Exemplarily, only the results using feature set BFCC2MFCC_xyzm are presented. These show the general trend that a sampling rate of 100 yields the best results.

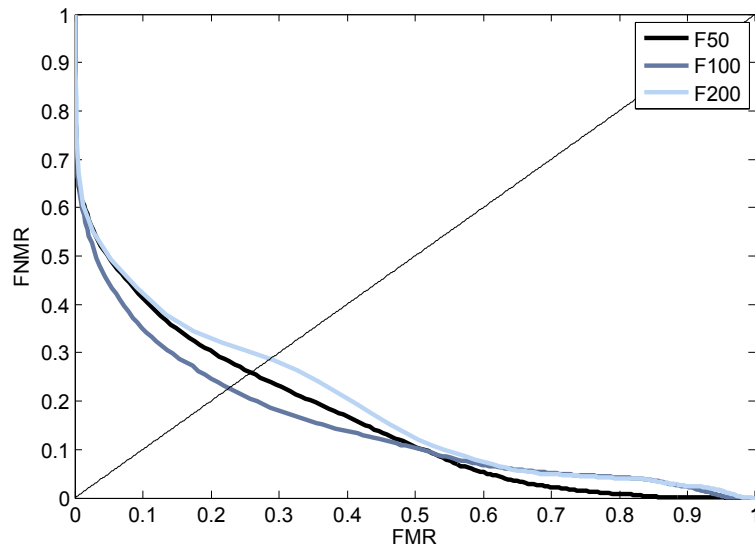


Figure 7.20.: DET-curves obtained when different sampling rates are used during cycle extraction. The cycles are normalized to a fixed length of 100 samples and feature vector BFCC2MFCC_xyzm is extracted from each cycle.

7.5.2. Varying the Normalized Cycle Length

Now, the sampling rate is set to 100 Hz and the cycle length (number of data values) is varied. In the normalization step of the cycle extraction the length is set to 50, 100, or 200 samples. The results using feature set BFCC2MFCC_xyzm are given in Figure 7.21. There is no big difference between cycle length 50 and 100. Although cycle length 100 seems to be a bit better, the results for a length of 50 samples are stated from now on as this yielded better results in the following evaluations and also decreases the computational effort.

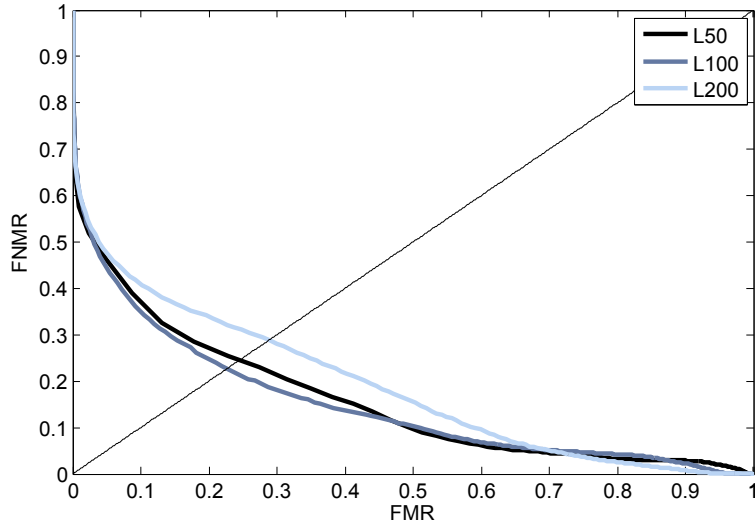


Figure 7.21.: DET-curves obtained when the cycles are normalized to different lengths. The data were interpolated to a fixed rate of 100 Hz before.

7.5.3. Concatenating Cycles

So far, the features were calculated for each cycle separately. This means, that each feature vector represents about one second of accelerometer data. As this might be a too short duration, up to five cycles are concatenated and the features are extracted from the resulting data set. If the total number of cycles is not divisible by the number of cycles to concatenate, the remaining cycles are omitted. The influence of this number of cycles is depicted in Figure 7.22. One can see that increasing the number of concatenated cycles from one to four does gradually decrease the EERs from 24.66% to 17.96%. Because the error rates get worse when five cycles are used (EER = 19.45%), four cycles were identified as the best configuration. When increasing the number of cycles to concatenate, the amount of cycles which are omitted at the end is also increased. This is probably the reason for the increased EER when five cycles are used. At this point the advantage of extracting the features from a larger amount of data does not compensate anymore for the loss of cycles.

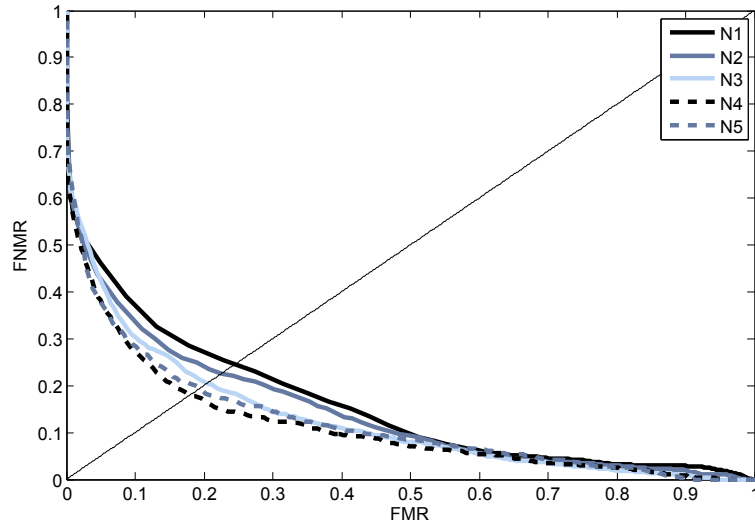


Figure 7.22.: DET-curves obtained when different numbers of cycles are concatenated before cycle extraction. The number of cycles varies from one to five. Results are given for feature vector BFCC2MFCC_xyzm.

7.5.4. Comparison of Different Features

For the previously stated best preprocessing configuration (sampling rate 100 Hz, cycle length 50, and grouping four cycles) now the results for different feature vectors are compared. The usual approach in cycle-based methods is to directly use the raw data as feature vector. Therefore, the initially created feature vector contains the raw data, separately for each axis. With the given configuration, each of these vectors has 200 entries. The obtained EERs are given in Table 7.18.

axis	x	y	z	m	xm	ym
EER	32.47	32.98	35.09	29.78	31.71	30.44

Table 7.18.: EERs obtained for the raw cycle data of the different axes and the magnitude vector (sampling rate 100 Hz, cycle length 50 and grouping four cycles).

The error rates are nearly twice as high as the so far stated ones for feature set BFCC2MFCC_xyzm, which shows that the feature extraction is a necessary step. One can see that the side-way acceleration (z-direction) has the least discriminative power. The x-direction and the magnitude vector perform best. This corresponds to the findings in the previous sections. Therefore, it is analyzed if using only the features from these two directions yields lower error rates than the so far used combination of all three axes and the magnitude vector. Table 7.19 gives the results for the three best single features. It shows that the so far presented feature vector BFCC2MFCC_xyzm performs best.

Feature Set	xyzm	x	m	xm
BFCC1	21.45	27.56	21.70	21.43
BFCC2	18.95	29.06	20.42	18.74
MFCC	18.91	26.03	22.20	20.44
BFCC2MFCC	17.96	25.77	21.03	18.94

Table 7.19.: EERs for the three best single features and the best combination (sampling rate 100 Hz, cycle length 50 and grouping four cycles).

7.5.5. Voting

These error rates can be further decreased by applying the voting mechanism. The maximal number of votes $\#V$ is 14, as this is the minimal number of available probe data of a subject when four cycles are concatenated. Figure 7.23 gives the EER for different combinations of $\#V$ and $\#GV$ for feature vector BFCC2MFCC_xyzm. For all $\#GV$, best results are obtained when grouping the maximal number of 14 votes. The results for $\#GV = 1$ to $\#GV = 4$ are all similar, using $\#GV = 3$ yields the highest decrease of the EER from 17.96% to 15.46%.

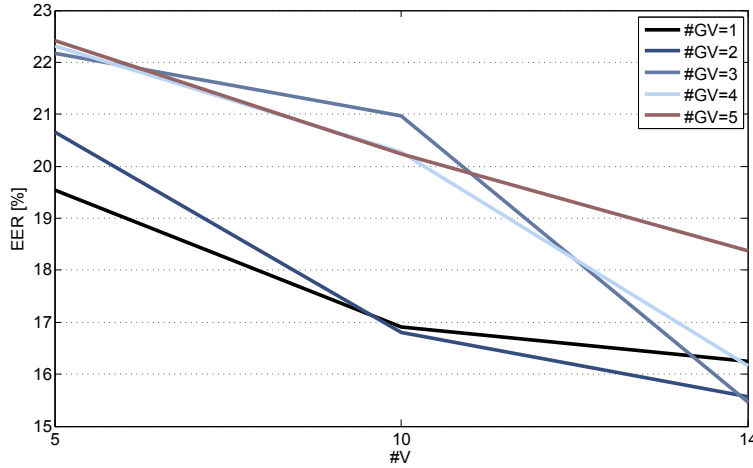


Figure 7.23.: EERs for different combination of $\#V$ and $\#GV$ when using feature vector BFCC2MFCC_xyzm (sampling rate 100 Hz, cycle length 50 and grouping four cycles).

7.5.6. Benchmarking Cycle-based against Fixed-length Time Segmentation

The goal of this evaluation was to see if a cycle-based segmentation results in lower error rates than a fixed-length segmentation which does not consider the characteristics of the signal. HMMs are used for classification and the results now will be benchmarked to the ones obtained in Section 7.2. The benchmark is based on the results obtained when using only data of setting 1 for training and testing³. For the fixed-length results the data were interpolated to a sampling rate of 50 Hz

³In [105] the stated HMM results are obtained when training with setting 1 and 3.

BFCC2MFCC_xyzm	cycle-based	fixed-length
before voting	17.96	17.28
after voting	15.46	13.89

Table 7.20.: Comparison of the EERs for cycle-based and fixed-length segmentation.

and divided into segments of five seconds (best performing configuration).

The optimal identified voting configuration for the fixed-length time segmentation was $\#GV = 4$ and $\#V = 30$. For the cycle-based segmentation the best voting configuration is $\#GV=3$ and $\#V=14$. For both segmentations the same feature vector BFCC2MFCC_xyzm performs best. Table 7.20 gives the results for both segmentations. Before voting, the results are similar. But applying the voting algorithm decreases the EER stronger when the fixed-length time segmentation is used. The reason for this might be the lower amount of votes $\#V$, which can be used when the cycle-based segmentation is applied. Using probe data of a longer duration and being able to increase $\#GV$ might eliminate this difference. Nevertheless, the comparison of the results without voting shows that using a cycle-based segmentation does not improve the error rates. Hence, the additional effort of the cycle extraction is not necessary.

7.6. Benchmark with Published State-of-the-Art Method

The gait recognition approach developed by J. Frank [47] was already presented in Section 3.4.3. Frank published his code⁴ which made it possible to evaluate his method on the database used in this chapter. In his publication [47] he presented the results obtained on a database collected using a G1 phone. Models are trained for 25 subjects and the test data are classified as belonging to the subject for whose models the highest similarity is obtained. Hence, he is aiming at an identification, whereas the approaches presented in this thesis accomplish the verification. In this evaluation Frank’s approach is used to get results comparable to the ones stated in his paper. To be comparable to the results presented in the previous sections of this chapter, in addition a threshold-based analysis is performed.

7.6.1. Data Preparation

Frank’s database contained only between twelve and 120 seconds of data for 25 subjects. The sampling rate was around 25 Hz. Now, setting 1 of database 2 is used for evaluation, containing between 8.9 to 16.9 minutes of data for 36 subjects, collected on two different days. The sampling rate is around 127 Hz. The publication does not state which acceleration direction is used for the evaluation. Therefore, the x-acceleration is used here, which was the best single axis in the previous evaluations. To prepare the data, for each subject all x-accelerations of the first session of this subject are concatenated to get the training data. To get the testing data, for each subject all x-accelerations of the second session of this subject are concatenated.

⁴<https://github.com/jwf/tdeotools>

7.6.2. Evaluation

The published code contains mainly two functions. *BuildTree* is used to train the models, *ClassifyTrajectory* is used for classification. One model is created per subject using the prepared training data of the first session. During testing, each test data file (one per subject) is mapped to each model space⁵, resulting in 36 classifications per subject. A sliding-window approach is used to get one similarity score for each segment of length 32 data values. Therefore, the result of a single comparison between the data of subject i and the model of subject j is a vector of similarity scores, which is only 31 elements shorter than the testing file.

Parameter Selection

The goal of this evaluation is to determine the optimal parameter configuration for the used data. As in Frank's publication, the similarity score vectors are converted into one single similarity score by taking the mean of the values. This yields 36 genuine scores (one for each subject) and 35×36 impostor scores, because the probe data of each subject is compared to all 35 models of the other subjects. Based on the scores, the FMR and FNMR is calculated for various thresholds and the EER is determined.

Three parameters can be set during training. The embedding dimension M , the reduced dimension P used when applying PCA, and the delay D . For a detailed description of the parameters see [47]. The configuration $M = 7$, $P = 7$, and $D = 2$ is used in an initial evaluation to determine a suitable sampling rate. Table 7.21 shows the obtained EERs.

Sampling Rate [Hz]	10	15	25	50	100
EER [%]	13.85	12.16	13.97	18.59	18.25

Table 7.21.: EERs obtained for different sampling rates.

The best sampling rate of 15 Hz was used to analyze the influence of the training parameters. M and P were chosen to be in $\{1, 3, \dots, 15\}$, and D was set to 1, 2, or 3. Table 7.22 shows a selection of the results. The lowest EER of 10.95% is obtained with $M = 5$, $P = 5$, and $D = 1$. It was verified that the chosen sampling rate still performs best for the new parameter set. For sampling rate 10 Hz the EER is 14.13% and for 25 Hz an EER of 13.57% was obtained. Using a sampling rate of 15 Hz means that each segment of length 32 corresponds to around 2.1 seconds.

7.6.3. Benchmark to Frank's Results

In his publication Frank stated that he obtained a 100% GMR on his database. Now it is analyzed how the corresponding performance is on the database used in this chapter. Figure 7.24 gives the results in similar manner as Frank does. A matrix S is constructed containing the mean of the scores obtained when comparing the probe data of subject i with the model of subject j at position (i, j) ($i, j = 1, 2, \dots, 36$). Hence, the genuine scores are contained in the diagonal of the matrix. From this matrix the difference matrix F is determined by calculating the max $m(j)$ of each column and setting $F(i, j) = \text{abs}(S(i, j) - m(j))$. This matrix F is plotted in Figure 7.24, white corresponds to no difference. For a correct identification the entry on the diagonal should be white, meaning that the maximal similarity is obtained in the genuine comparison. Eight

⁵The default configurations are used for classification.

M\P	D=1				D=2			
	1	3	5	7	1	3	5	7
1	39.52	–	–	–	39.52	–	–	–
3	47.06	16.88	–	–	45.34	15.28	–	–
5	34.25	19.07	10.95	–	38.41	20.02	11.29	–
7	39.64	16.99	10.99	12.58	40.20	24.96	13.31	12.16
9	37.28	20.38	15.62	12.76	36.92	23.39	16.61	11.17
11	34.31	18.31	13.71	13.97	37.20	25.14	18.17	13.25
13	31.01	18.17	13.65	14.17	37.50	22.00	16.51	12.02

M\P	D=3			
	1	3	5	7
1	39.52	–	–	–
3	39.52	22.12	–	–
5	42.20	16.83	14.05	–
7	43.06	15.91	13.95	13.89
9	44.38	20.77	15.20	14.19
11	42.16	20.95	18.49	16.63
13	42.68	23.37	18.25	17.18

Table 7.22.: EERs for various combinations of training parameters M , P , and D .

classification errors occur, which are marked by a red frame in the figure. This corresponds to a GMR of 78.78%. Hence the perfect classification could not be confirmed on this database.

7.6.4. Benchmark to Results of this Chapter

Because of the different approach of considering the mean over all scores for evaluation, the so far obtained results for Frank's code are not comparable to the ones stated previously in this chapter. Now, the number of scores is reduced, to correspond to the 50% overlap of the segments. Each of the remaining scores contributes to the calculated FMRs and FNMRs. The resulting EER is 19.51%. This is higher than the best results stated for the approaches presented previously, although it was given attention to identify the best algorithm configuration for the applied database.

The voting algorithm is applied to evaluate if the EER can be further reduced. The number of genuine votes $\#GV$ was varied from 1 to 10 and the number of grouped votes $\#V$ was in the set $\{10, 20, \dots, 200\}$. Figure 7.25 shows the DET-curves obtained without voting and using the best voting configuration of $\#GV=9$ and $\#V=90$. Voting reduces the EER to 10.99%. Although this is much lower than the result without voting, the biometric performance is not as low as the one obtained for SVMs and the k-NN algorithm evaluated in this chapter.

7.7. Summary and Conclusions

This chapter presented the evaluation results for three different machine learning algorithms, namely HMMs, SVMs, and k-NN. For all three algorithms the evaluation followed similar steps, to

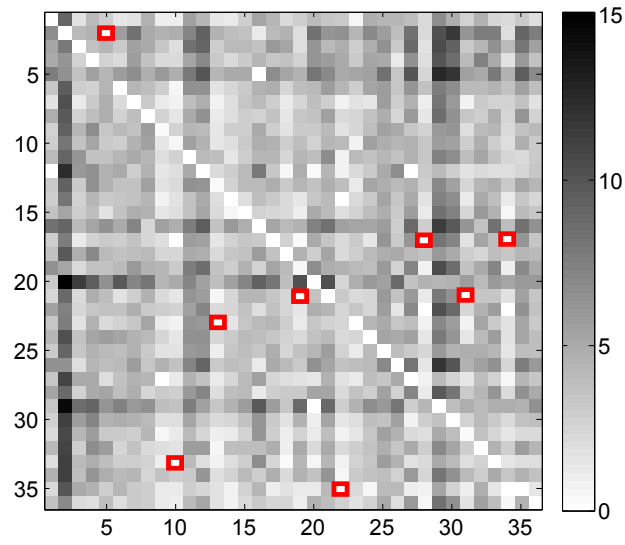


Figure 7.24.: Identification results using Frank's method. For each column, distances between maximal score of that column and the other scores are plotted. Classification errors are marked with a red frame.

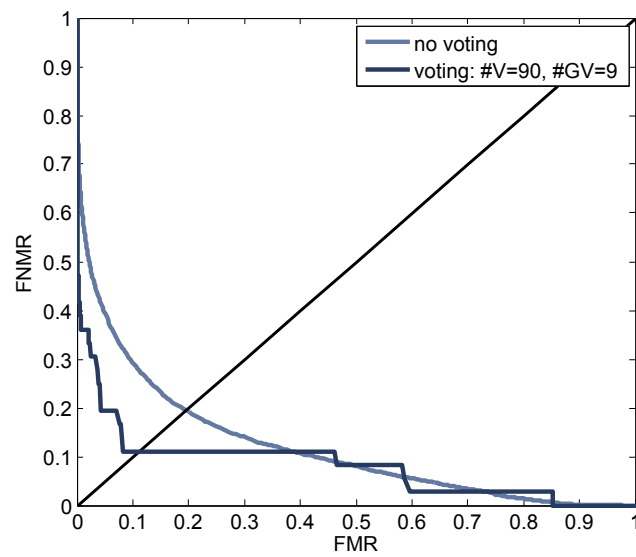


Figure 7.25.: DET-curves obtained with Frank's method without voting and with the best voting setting of $\#GV = 9$ and $\#V = 90$.

enable a benchmark of the results. Table 7.23 gives an overview over the best obtained EER or HTER for each classifier and each setting. For each setting, the results of the three classifiers are very similar. Therefore, none of these can be clearly preferred or determined to be unsuitable for accelerometer-based biometric gait recognition. For fast walking, extracting features from x-axis and magnitude vector and using the Bark scale gives best results. For normal walk there is no common best feature set.

A published state-of-the-art method has been evaluated on database 2 used in this chapter. Benchmarking the results to the ones achieved by the methods contributed in this thesis, shows that the state-of-the-art method is outperformed.

	Feature Set	Setting 1	Setting 3	Setting 2	
		EER/ HTER	EER/ HTER	Feature Set	EER/ HTER
HMM	BFCC2MFCC_xyzm	13.89	8.47	BFCC1_xm	8.24
SVM	BFCC1BFCC2_xm	8.68	10.93	BFCC2_xm	6.61
k-NN	BFCC2_xyzm	9.44	7.64	BFCC1_xm	9.72

Table 7.23.: Lowest obtained EER or HTER for each of the three evaluated algorithms (after voting).

The following five research questions are considered in this chapter:

Research Question 2: What are suitable features to describe subject specific gait? For a gait-based classification of subjects, BFCCs and MFCCs have been identified as suitable features. Statistical features like mean or standard deviation should not be used. DPS was applied to identify best feature combinations, containing statistical as well as BFCC and MFCC features, but the results did not outperform those of the cepstral coefficients alone. Which combination of BFCC1, BFCC2, and MFCC was best, varied for each classifier and velocity. In each case, extracting the features from all three axes and the magnitude vector or only from x-axis and magnitude vector was most suitable.

Research Question 3: How much training data are necessary when applying machine learning algorithms to accelerometer-based biometric gait recognition? Containing the large amount of around 30 minutes of walking data per subject, the database allowed for an evaluation of the influence of training data to the classification performance. For HMMs and the k-NN algorithm, increasing the amount of training data did result in a close to monotone decrease of the error rates, being stronger in the beginning and being less from around 3 minutes of training data. The amount of training data should be as large as possible, but at least 4.5 minutes. For SVMs the behavior of the error rates is slightly different. In general, the effect of increasing the amount of training data is similar to that of the other two algorithms. But outliers may occur, which result in very high FNMRs. If this occurs during enrolment, the created classifier is unsuitable and the enrolment has to be repeated, resulting in a lower usability.

Research Question 4: Does the biometric performance decrease with aging of the reference data? By opposing same-day and cross-day performances it has been shown that aging of the template does decrease the biometric performance. With increasing time be-

tween storing the reference and recording the probe there is a higher possibility that several covariates (shoes, trousers, mood) change which have an influence to the gait.

Research Question 5: How is the biometric performance when reference and probe are obtained from data collected with different walking velocities? It has been shown that there is a high degradation of the EERs when reference and probe are collected under different walking velocities. Best results are obtained when probe and reference correspond to the same velocity, but also training the machine learning algorithms with mixed-velocity data gives acceptable results.

Research Question 7: Are current smartphones capable of performing the necessary feature extraction and classification steps with an acceptable computational complexity? The k-NN algorithm has been implemented as MBASSy module. The determined times for feature extraction, training, and classification are low enough to use this module for authentication on smartphones.

CHAPTER 8

CHALLENGING EVALUATION ON A REALISTIC DATABASE

The main research question considered in this chapter is: *How is the biometric performance of the proposed methods in a scenario test?* Furthermore, most other research questions are considered, giving a comprehensive evaluation of the different developed gait recognition methods on a realistic database. HMMs, SVMs, and the k-NN algorithm are used to classify feature sets consisting of various combinations of MFCCs and BFCCs, similar to the evaluation in the previous chapter. In addition, the cycle extraction method, presented in Section 5.2.1, is evaluated using the same database. This evaluation was directly performed on the mobile phone to evaluate the required enrolment and authentication runtimes. Parts of this chapter have been published in [110] and [107].

8.1. Challenging Data Collection Using Motorola Milestone

48 subjects took part in the data collection for database 3. See Table 8.1 for age and gender distribution. The participants were told to walk in their normal pace during the whole test. Each participant took part twice on two different days and in most cases was wearing the same shoes during both sessions. The phone was inside a pouch which was attached on the right side of the hip of the subject.

	20-24	25-29	30-39	40-49	50+	total
male	7	11	8	3	1	30
female	7	6	2	1	2	18
total	14	17	10	4	3	48

Table 8.1.: Age and gender distribution of participants of the third data collection.

The data collection started with two short walking periods of 10 seconds, in which the subjects only walked straight on a flat floor. Start and end of these periods were indicated via an acoustical signal from the phone. The raw data were stored to be used in further evaluations.

Afterwards the subjects had to walk on a predefined route three times. This route involved two floors of a building which has a rectangular shape with a patio in the center, which allows finding a route without dead end. The route is depicted in Figure 8.1. During walking on that route the subjects had to stop at nine predefined points approximately 25 seconds apart from each other. To not influence them, the subjects were walking unattended but there was one person standing at each point to stop and restart the data collection. The different sections will be evaluated separately later.

The route was chosen in a way such that it corresponds to a realistic scenario. In former data collections, also those used in state-of-the-art publications, subjects mainly walked on a straight line. In this case, the setting is more advanced and contains walking around corners (left and right), walking stairs up and down, opening and closing doors, and having to cross door sills. Figure 8.2 shows the route section from authentication point 5 to 6. One can see the two different kinds of floor the subjects walked on (linoleum and tiles), the door sills which occurred several times during the route (also indicated in Figure 8.1), and chairs which prevented the subjects from walking on a straight line. Figure 8.3 shows the area around the starting point. Between authentication point 3 and 4 the subjects had to open the glass door and walk upstairs. In the last section (between authentication point 8 and 9) this door had to be opened and closed by the subjects, which forced them to stop.

During the evaluation these different parts of the data collection are referred to as *setting*. For example, data of setting 1 are collected between the start and authentication point 1, data of setting 2 between authentication points 1 and 2, etc. Setting 0 (enrolment data) corresponds to the short walking period at the beginning. The data collected between start and end point are called *walk*. Because the subjects walked three times on the route, three walks are obtained. Hence, for each of the 48 subjects two enrolment data sets and 27 authentication data sets, three for each setting, were obtained in each session, 2784 in total.

For the evaluations, the data are divided into segments of 2, 3, 4, 5, 6, and 7.5 s, with an overlap of 50%. Table 8.2 shows the median number of resulting segments per setting as well as the mean duration of each walk.

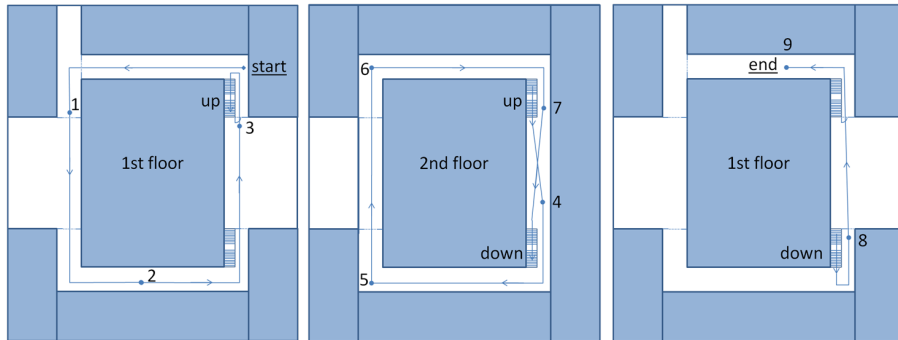


Figure 8.1.: Subjects walked on this route. Dashed lines indicate door sills.



Figure 8.2.: Area in the upper floor between authentication point 5 and 6.



Figure 8.3.: Area around the starting point which is crossed several times.

	Setting								
	1	2	3	4	5	6	7	8	9
Duration	22.6	26.7	26.3	28.0	25.5	24.3	22.1	25.9	21.6
2 s	21	25	25	27	24	23	20	24	21
3 s	13	16	16	17	15	15	13	16	13
4 s	10	12	12	13	11	11	11	11	10
5 s	7	9	9	10	9	8	7	9	7
6 s	6	7	7	8	7	7	6	7	6
7.5 s	4	6	5	6	5	5	4	5	4

Table 8.2.: Mean duration of each setting (in seconds) and the median number of segments after segmentation to 2, 3, 4, 5, 6, and 7.5 s.

8.2. Hidden Markov Models

Due to the promising results of HMMs, they were also evaluated on this more challenging database. Again, the preprocessing involved linear interpolation to sampling rate 25, 50, and 100 samples per second, centering around zero by subtracting the mean, and dividing the data into overlapping segments of length 2, 3, 4, 5, 6, and 7.5 s. Afterwards, features are extracted from each segment, separately for the x-, y- and z-direction as well as for the magnitude vector m . Because statistical features, like standard deviation, minimum or maximum, have not resulted in sufficient low error rates when testing HMMs on database 2, this evaluation focuses on MFCCs and BFCCs.

8.2.1. Evaluation

Different tests are conducted which are described in the following sections. A corresponding evaluation was published in [107]. For this section the evaluation was extended by further preprocessing configurations and additional feature sets¹. If not mentioned differently, all results are obtained using setting 0 to 9 of the first walk at the first session for training and using setting 1 to 9 of the first walk of the second session for testing.

Best Performing Feature Set

Table 8.3 gives the EERs obtained when extracting the features for each axis and combining the results. All results range between an EER of 17.68% and 18.95%; hence there are no significant differences between the tested feature sets. Feature set BFCC2MFCC_xyzm is chosen for the following evaluations as it already showed good results for database 2. Figure 8.4 shows the results when using feature set BFCC2MFCC for different acceleration directions and combinations. When using only one axis, the vertical acceleration x and the magnitude vector m yield best results. Combining these two into feature set BFCC2MFCC_xm further decreases the error rates, but best results are obtained using BFCC2MFCC_xyzm. Full results are given in Appendix E.

Feature Set	Vector Length	EER
BFCC1_xyzm	52	18.83%
BFCC2_xyzm	52	17.74%
MFCC_xyzm	52	17.80%
BFCC1BFCC2_xyzm	104	17.93%
BFCC1MFCC_xyzm	104	17.68%
BFCC2MFCC_xyzm	104	17.31%
BFCC1BFCC2MFCC_xyzm	156	17.42%

Table 8.3.: EER for different feature sets at sampling rate 50 Hz and segment length 2 s.

Optimal Preprocessing Configuration

The previously stated results are obtained when using a sampling rate of 50 samples per second during preprocessing and a segment length of 2 s. Table 8.4 shows the results for feature set BFCC2MFCC_xyzm using sampling rate 25, 50, and 100 Hz and segment lengths between 2 and

¹Slight variations of the error rates are due to a small change in the segmentation algorithm.

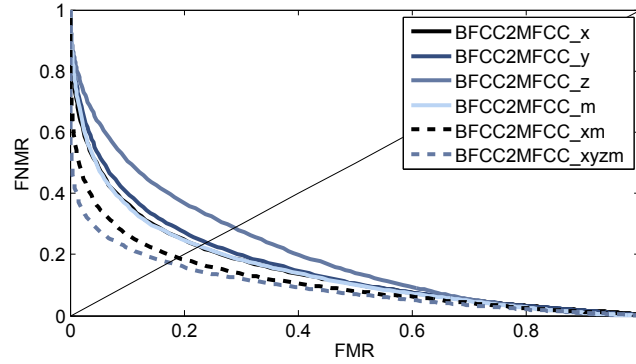


Figure 8.4.: DET-curves for feature BFCC2MFCC obtained for different axes and axes combinations at sampling rate 50 Hz and segment length 2 s.

7.5 s. Full results for all feature sets are given in Appendix E. One can see that sampling rates 50 and 100 Hz yield nearly the same results and sampling rate 25 Hz is slightly worse. This confirms the correct choice of sampling rate 50 Hz, because a lower sampling rate decreases the computational effort. Although higher segment lengths yield slightly better results, results for segment length 2 s are presented in the following evaluations because the short segments will get lower error rates when applying the voting algorithm.

BFCC2MFCC_xyzm	2 s	3 s	4 s	5 s	6 s	7.5 s
25 Hz	18.00	16.87	16.32	15.86	15.59	15.04
50 Hz	17.31	16.11	15.63	15.23	15.17	14.88
100 Hz	17.46	16.21	15.36	15.11	15.08	14.86
200 Hz	17.55	16.17	15.35	14.95	15.02	14.82

Table 8.4.: EERs for sampling rate 25, 50, and 100 Hz and different segment lengths.

Optimal Amount of Training Data

Figure 8.5 shows the EERs obtained for different amounts of training data. Ten different random sequences (see Table E.1 in Appendix E) are used to obtain results which are not influenced by the order of the settings. In addition, the results for the original sequence and the median of all results are given. The figure shows that the error rates are highly influenced by the sequence and no common behavior can be observed. The median decreases fast from 21.04% to 17.31% when using three instead of one setting for training. The lowest EER of 17.16% can be observed for eight training settings, corresponding to around four minutes of data and this amount is considered the minimal time necessary for enrolment. Adding more data does not further decrease the error rates. Using all nine settings yields an EER of 17.29% and adding also setting 0 to the training data gives an EER of 17.46%. Only data collected at the first walk were considered so far. Training the models with setting 1 to 9 of walk 1 and 2 yields an EER of 17.33% and using setting 1 to 9 of all three walks keeps the EER at the same level (17.57%).

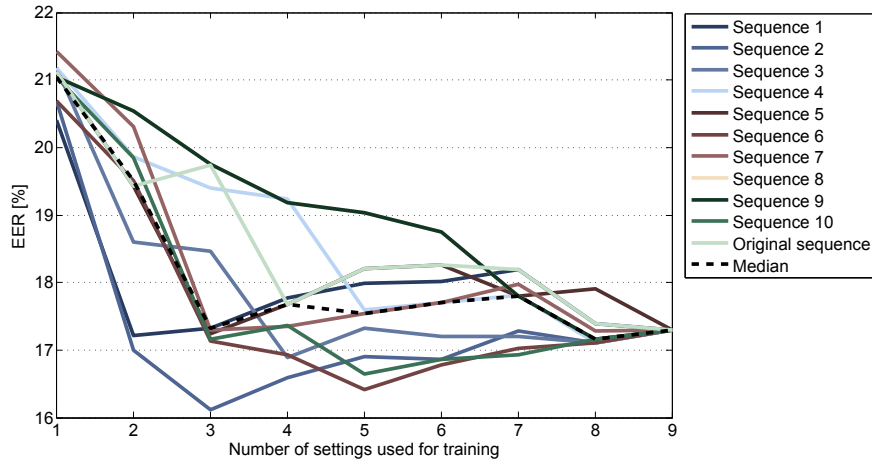


Figure 8.5.: EERs for different amounts of training data (sampling rate 50 Hz, segment length 2 s, feature set BFCC2MFCC_xyzm).

Separate Results for Each Test Setting

The system was trained using data from sections 0 to 9 collected on the first day. Now the performance is analyzed, when using the settings separately for testing. Figure 8.6 shows the obtained EERs as bar plots. One can see that the results are similar for all settings; only setting 4 and 8 give worse results. This is due to the fact that these are the settings where the subjects partly had to walk on stairs. As no activity recognition has been done during preprocessing to identify the walking parts, the data recorded while the subjects are walking up-stairs or down-stairs are still included. Therefore, some probe segments do not contain walking data and hence worsen the results. Also setting 9 yields slightly worse error rates, which is due to the fact, that opening and closing the door did distract the walk of the subjects.

Same-day Results

To analyze the influence of time on gait, same-day results are computed. The training set consists as before of data from settings 0 to 9 collected during the first walk at the first day. Now the test data consists of settings 1 to 9 from the second walk of the first session. In this case an EER of 10.66% for feature set BFCC2MFCC_xyzm is obtained (segment length 2 s, sampling rate 50 Hz), which is much less than the so far reported cross-day result. This indicates the high variability of gait over time and shows the importance of conducting research on a database that was created on two different days.

Subject-wise Results

A further interesting aspect is the stability of the results over all subjects. The error rates are more reliable if they do not vary much for different subjects. When analyzing the cycle extraction results in Section 8.5 a high variability can be observed. Figure 8.7 gives the results for HMMs using feature set BFCC2MFCC_xyzm. For each subject the FNMR is plotted which is obtained at

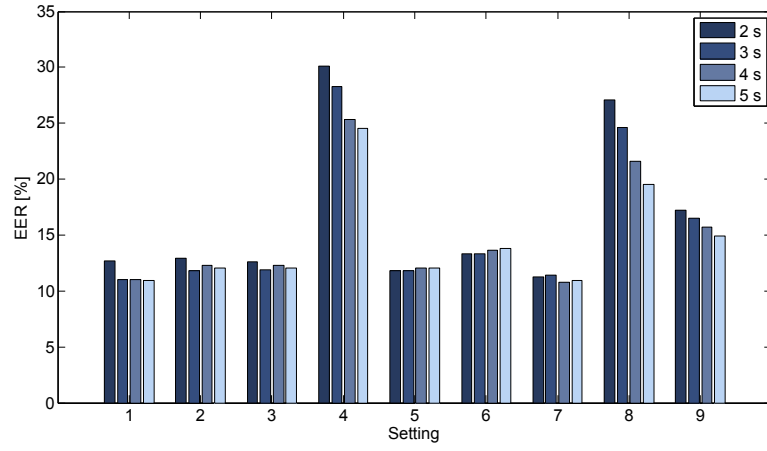


Figure 8.6.: EERs separately for each setting using feature vector BFCC2MFCC_xyzm with segment lengths 2, 3, 4, and 5s.

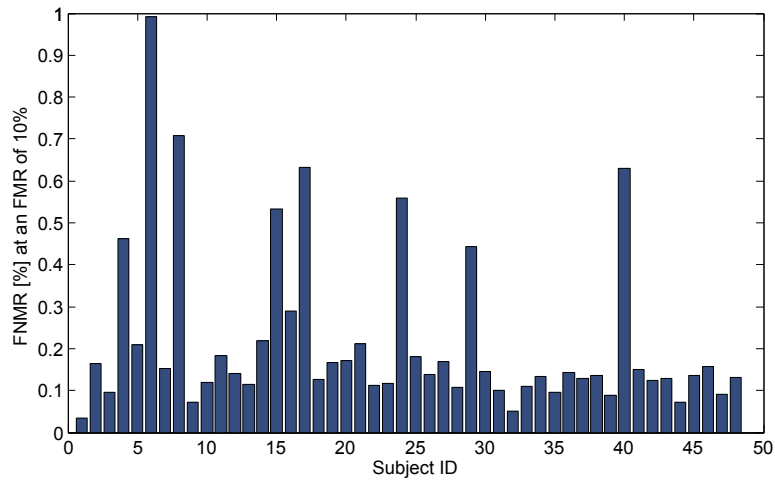


Figure 8.7.: FNMR for each of the 48 subjects at an FMR of 10% for segment length 2s and sampling rate 50 Hz.

an overall FMR of approximately 10%. One can see that the FNMR is around 13% for most of the subjects, with only eight outliers. Hence, a much higher stability than for the cycle extraction method is obtained. The reason for some of the outliers might be that a few subjects (IDs 8, 13, 15, 17, 27) wore different shoes on the two sessions. But also the position of the phone-pouch due to different trousers could be a reason.

Voting

Until now, each classification result for each segment is directly used to calculate the FMR and FNMR. This section presents the voting results. Figure 8.8 shows the results obtained for feature BFCC2MFCC_xyzm and different numbers of votes $\#V$ and genuine votes $\#GV$. Figure 8.9 gives the DET-curves without voting and for different $\#V$ and $\#GV = 1$. Without voting an EER of 17.31% was obtained. Using $\#V=100$ and $\#GV=1$, these error rate can be decreased to 6.13%. Because segments of length 2 s are considered in this evaluation (which overlap by 50%) these 100 votes correspond to 101 seconds of data used for classification.

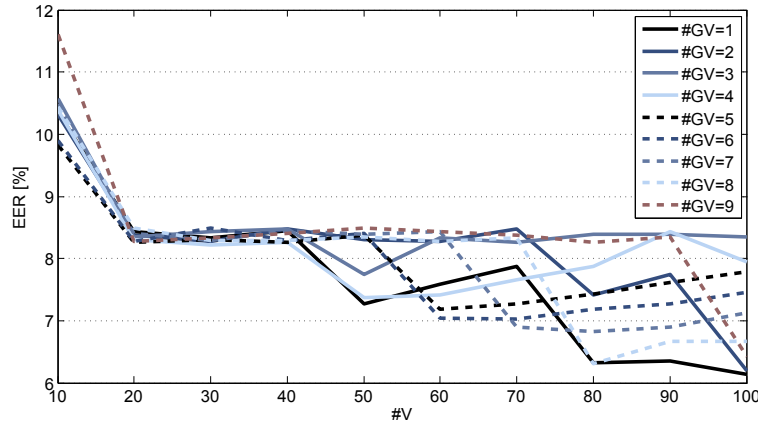


Figure 8.8.: Influence of the voting scheme (feature set BFCC2MFCC_xyzm, segment length 2 s, sampling rate 50 Hz, trained using setting 0 to 9).

8.3. Support Vector Machines

From the comparison of the performance of the machine learning algorithms on database 2 could not be inferred which of the approaches is to be preferred. Therefore, SVMs were evaluated in a similar manner as HMMs on this more challenging database.

8.3.1. Evaluation

The evaluations were performed for all combinations of features BFCC1, BFCC2, and MFCC. All axes and the magnitude vector were used separately. In addition, the combination of x-axis and magnitude vector and the combination of all three axes and the magnitude vector were used. Results for all feature sets, all segment lengths, and all sampling rates are given in Appendix F.

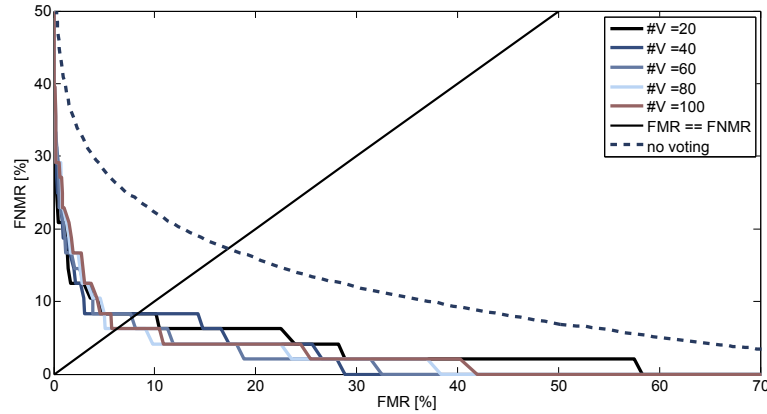


Figure 8.9.: DET-curves for different $\#V$ and $\#GV = 1$ and without voting (segment length 2 s, sampling rate 50 Hz, feature set BFCC2MFCC_xyzm, trained using setting 0 to 9).

Best Performing Feature Set

Table 8.5 shows the error rates obtained when using sections 0 to 9 for training, a sampling rate of 50 samples per second, and a segment length of 5 s. Analyzing the results obtained for the three different axes, one can see that acceleration in vertical direction (x-axis) yields much better results than horizontal and forward-backward acceleration. The results using the magnitude vector are similar to the ones when using the x-axis. Therefore, additional feature sets were created, combining the features from x-axis and the magnitude vector. Because this combination outperformed the so far best single feature, further feature sets containing all three axes and the magnitude vector were evaluated. Best results are obtained using the combination of x-axis and magnitude vector. Best feature set is BFCC1MFCC_xm, yielding an FMR of 2.43% at an FNMR of 32.40% (HTER = 17.41).

Best Preprocessing Configuration

Previously, a sampling rate of 50 samples per second and a segment length of 5 s were used. Table 8.6 shows the results for various additional sampling rates and segment lengths obtained when using setting 0 to 9 for training.

One can see that there is no high influence of the sampling rate to the error rates. The best results for each segmentation length are obtained for sampling rate 50 Hz. For each sampling rate, the best result is obtained using segment length 5 s or 6 s. The best combination is a sampling rate of 50 Hz and a segment length of 5 s, as used in the previous evaluations.

Optimal Amount of Training Data

To identify the optimal amount of training data, the same ten random sequences as in the HMM evaluation are applied. Starting with one setting for training, successively all other settings were added to the training set. Test data consisted in all cases of setting 1 to 9 collected on the second day during the first walk. Figure 8.10 gives the results for feature set BFCC1MFCC_xm, sampling

feature set	length	FMR	FNMR	HTER
BFCC1_x	13	4.56	46.48	25.52
BFCC2_x	13	4.19	45.93	25.06
MFCC_x	13	3.11	47.66	25.39
BFCC1BFCC2_x	26	3.36	46.29	24.82
BFCC1MFCC_x	26	2.54	46.89	24.72
BFCC2MFCC_x	26	2.63	48.54	25.59
BFCC1BFCC2MFCC_x	52	2.24	47.33	24.79
BFCC1_y	13	3.41	53.05	28.23
BFCC2_y	13	3.14	55.89	29.51
MFCC_y	13	2.77	57.67	30.22
BFCC1BFCC2_y	26	2.26	53.33	27.79
BFCC1MFCC_y	26	2.23	57.32	29.77
BFCC2MFCC_y	26	2.27	57.95	30.11
BFCC1BFCC2MFCC_y	52	2.35	52.92	27.63
BFCC1_z	13	4.44	62.51	33.48
BFCC2_z	13	4.24	59.08	31.66
MFCC_z	13	5.06	62.79	33.92
BFCC1BFCC2_z	26	2.93	60.67	31.80
BFCC1MFCC_z	26	2.85	64.16	33.50
BFCC2MFCC_z	26	2.74	62.32	32.53
BFCC1BFCC2MFCC_z	52	1.88	64.30	33.09
BFCC1_m	13	3.44	47.74	25.59
BFCC2_m	13	3.08	45.65	24.37
MFCC_m	13	2.64	48.32	25.48
BFCC1BFCC2_m	26	2.31	43.37	22.84
BFCC1MFCC_m	26	2.12	46.26	24.19
BFCC2MFCC_m	26	2.38	46.37	24.38
BFCC1BFCC2MFCC_m	52	1.73	47.61	24.67
BFCC1_xm	26	1.81	36.06	18.93
BFCC2_xm	26	1.91	36.55	19.23
MFCC_xm	26	1.87	37.54	19.71
BFCC1BFCC2_xm	52	2.53	33.99	18.26
BFCC1MFCC_xm	52	2.43	32.40	17.41
BFCC2MFCC_xm	52	0.92	42.30	21.61
BFCC1BFCC2MFCC_xm	78	1.98	33.58	17.78
BFCC1_xyzm	39	1.48	39.44	20.46
BFCC2_xyzm	39	1.86	38.92	20.39
MFCC_xyzm	39	1.74	37.73	19.74
BFCC1BFCC2_xyzm	78	2.61	40.07	21.34
BFCC1MFCC_xyzm	78	2.54	37.65	20.09
BFCC2MFCC_xyzm	78	2.66	38.20	20.43
BFCC1BFCC2MFCC_xyzm	117	1.79	37.95	19.87

Table 8.5.: SVM results for all different feature sets (sampling rate 50 Hz, segment length 5 s).

BFCC1MFCC_xm	25 Hz		50 Hz		100 Hz		200 Hz	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
2 s	3.77	41.72	3.37	37.77	3.46	37.88	3.71	38.18
3 s	2.95	38.37	2.97	36.10	3.17	35.81	3.10	36.74
4 s	1.31	40.03	1.13	40.05	2.62	35.83	2.59	36.48
5 s	1.01	40.18	2.43	32.40	2.39	35.25	0.96	42.74
6 s	0.72	42.73	0.55	43.15	2.22	35.56	0.61	45.68
7.5 s	2.15	33.85	1.95	33.06	2.06	35.97	2.32	35.50

Table 8.6.: FMR and FNMR for feature set BFCC1MFCC_xm using various sampling rates and segment lengths.

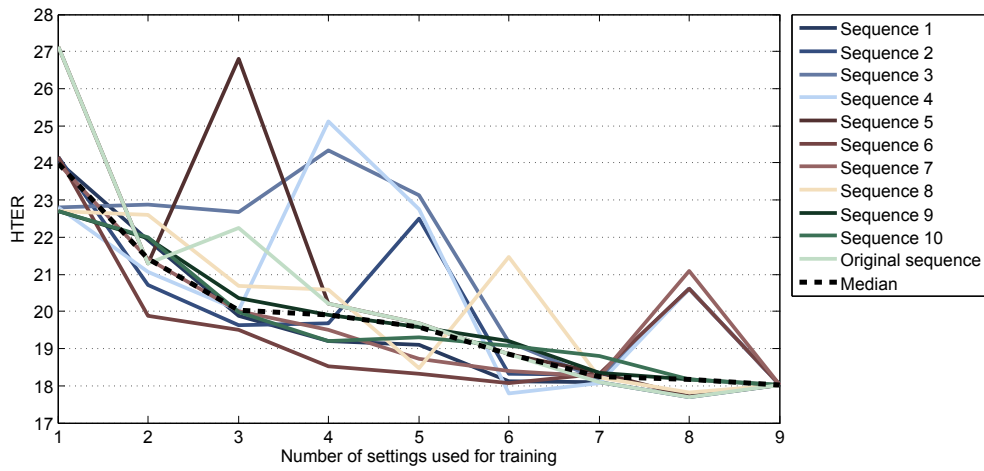


Figure 8.10.: HTER for different amounts of training data (sampling rate 50 Hz, segment length 2 s, feature set BFCC1MFCC_xm).

rate 50 Hz and segment length 5 s for the ten different random sequences, the original sequence and the median.

Like in the evaluation on database 2, several outliers can be observed which highly influence the recognition performance. No pattern can be detected of settings which result in this bad performance. Considering the median, the general trend is observable. Increasing the amount of training data from 1 to 3 settings yields a high decrease in the HTER from around 23.98 to 20.03. Adding additional data up to seven settings further decreases the HTER to 18.25. This corresponds to three minutes of walking and should be the minimal amount of training data. Adding further two settings does lower the HTER to 18.02 and the lowest HTER of 17.41 was obtained when using all settings (0 to 9) of the first walk for training. Training the SVMs with setting 1 to 9 of walk 1 and 2 yields an HTER of 18.58. Using setting 1 to 9 of all three walks for training results in an HTER of 18.01.

Separate Results for Each Test Setting

Similar to the HMM evaluation, each setting is used separately for testing. Segment length 5 s, sampling rate 50 Hz, and feature vector BFCC1MFCC_xm are used. Models are trained with data from all settings collected in the first walk at the first session. One test was performed for each setting. Each time testing data consisted of all walks collected during the second session for that setting. The results are given in Figure 8.11. For all settings they are on the same level, only settings 4, 8, and 9 are worse. This corresponds to the results for HMMs and is due to the fact that these settings contained the stairs or distraction by the door, respectively.

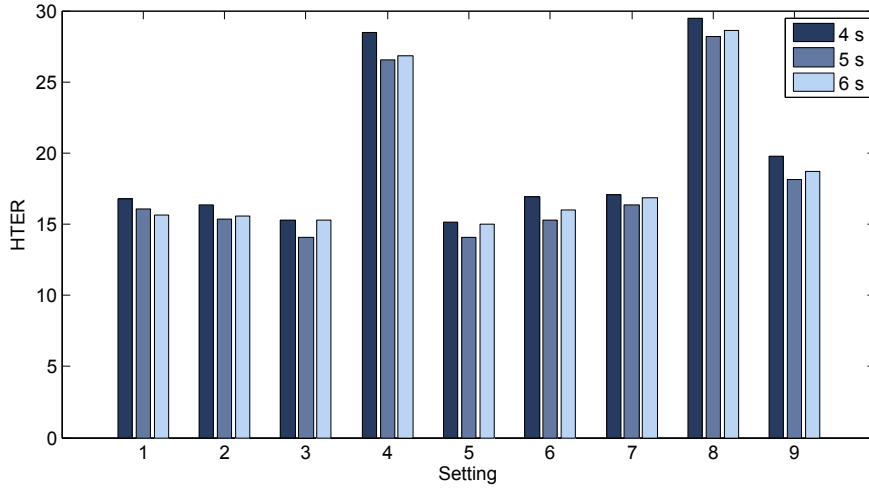


Figure 8.11.: HTER at each setting for different segment lengths (sampling rate 50 Hz, feature set BFCC1MFCC_xm).

Same-day Results

Same-day results were obtained by training the models with settings 0 to 9 (first walk), and testing with settings 1 to 9 of the second walk on the first session. This keeps the FMR on the same level as before (2.54% instead of 2.43%) and decreases the FNMR from 32.40% to 9.05% for feature set BFCC1MFCC_xm, segment length 5 s, and sampling rate 50 Hz.

Subject-wise Results

The stability of the results is analyzed by evaluating FNMR and FMR for each subject separately. The models were trained using data from setting 0 to 9 and tested using data from all settings of the first walk of the second session. There is a higher variability in the results than obtained using HMMs. But there are also similarities in the results of the two different classifiers. When using HMMs, high FNMR were obtained for subject 6, 8, 15 – 17, 24, 29, and 40 (see Figure 8.7) and these obtained also high FNMRs when using SVMs.

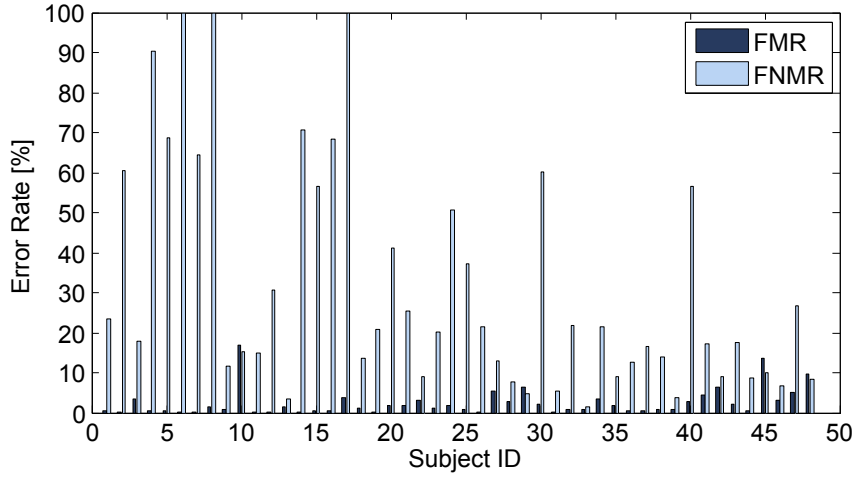


Figure 8.12.: FNMR and FMR for each of the 48 subjects (sampling rate 50 Hz, segment length 5 s).

Voting

Figure 8.12 shows that for most subjects the FNMR is far too high for a practical application. Therefore, the voting algorithm is applied to lower the FNMR. Varying the number of $\#GV$ and $\#V$ results in varying combinations of FMR and FNMR, which can be plotted as a DET-curve. Figure 8.13 gives the results obtained for $\#GV \in \{1, 2, \dots, 18\}$ and $\#V \in \{5, 10, 15, \dots, 60\}$. An EER of 6.38% is obtained for $\#GV=4$ and $\#V=55$, which is much lower than the previously stated HTER of 17.42 (FMR = 2.43% and FNMR = 32.40%). The 55 votes correspond to around 2 minutes walking. Table 8.7 shows the obtained HTER for different combinations of $\#GV$ and $\#V$. One can see that if the ratio of $\#GV$ and $\#V$ lies between 1:12 and 1:3 acceptable results are obtained.

	5	10	15	20	25	30	35	40	45	50	55	60
1	6.21	6.45	6.85	7.34	8.36	9.51	10.26	10.35	10.62	10.88	11.48	12.10
2	7.54	6.16	6.56	5.83	6.23	6.94	7.25	7.52	7.60	7.87	8.38	8.77
3	12.53	5.67	6.16	6.30	6.65	5.83	6.05	6.27	6.54	6.76	7.14	7.43
4	14.25	10.66	5.85	6.07	6.07	5.12	5.43	5.65	6.01	6.14	6.38	6.59
5	19.20	10.44	7.72	5.76	5.94	5.98	6.12	6.25	6.56	5.74	6.07	6.27
6	–	12.30	9.71	6.72	5.85	5.90	5.94	6.07	6.25	6.47	6.74	6.13
7	–	14.30	11.44	8.71	5.67	5.81	5.90	5.94	6.12	6.21	6.47	6.80
8	–	17.16	12.30	8.53	6.72	6.72	6.85	5.94	6.07	6.12	6.30	6.62
9	–	21.24	13.30	12.53	7.58	6.54	6.72	5.81	5.90	5.94	6.12	6.43
10	–	26.31	13.21	12.39	9.62	8.62	7.67	6.76	5.85	5.94	6.03	6.25

Table 8.7.: HTER obtained for $\#GV$ between 1 and 10 and $\#V$ between 5 and 60. Those values for which the ratio between $\#GV$ and $\#V$ is between 1:12 and 1:3 are marked gray.

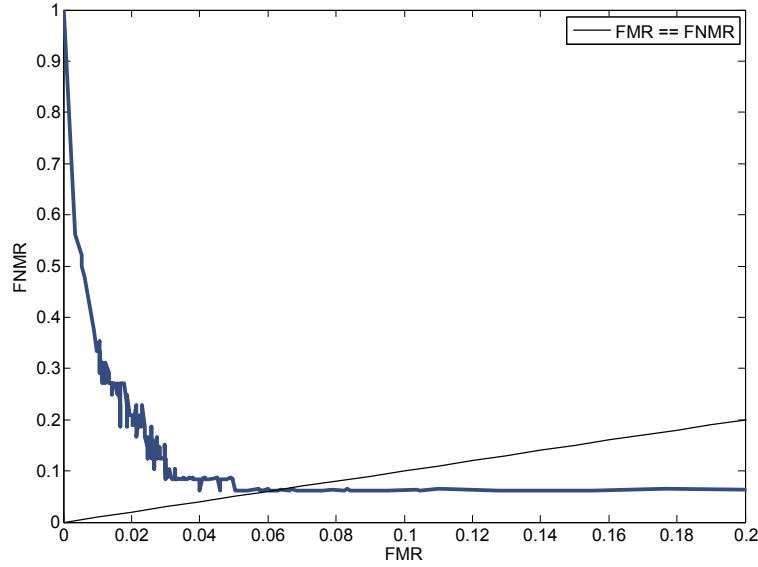


Figure 8.13.: DET-curve obtained by plotting FMRs and FNMRs obtained for different $\#GV$ and $\#V$.

8.4. k-Nearest Neighbor Algorithm

On database 2, the k-NN algorithm performed best. This evaluation analyzes if the biometric performance remains as good in the scenario test.

8.4.1. Evaluation

Because the evaluation in Section 7.4.1 showed that the influence of the number of neighbors is very low, this evaluation is performed considering three neighbors, as before.

Best Performing Feature Set

As before, the focus of the evaluation is on BFCC and MFCC. The results for all possible combinations of these features and different segment lengths and sampling rates are given in Appendix G. Table 8.8 states the results for segment length 7.5 s and sampling rate 50 Hz, obtained when using data of all ten settings for training. The two best performing feature vectors are MFCC_xyzm and BFCC2MFCC_xyzm. Due to the shorter length, the former is preferred. To analyze if it is possible to reduce the number of considered acceleration directions, these are analyzed separately. Table 8.9 states the error rates obtained when the feature vector consists of MFCCs computed for single axes and the combination of x-axis and magnitude vector. Using only single axes or the magnitude vector alone does not give acceptable results. The combination of magnitude vector and x-axis performs well but MFCC_xyzm stays the most suitable feature vector and is used in the following evaluations.

Feature Vector	Feature Length	FMR	FNMR	HTER
bfcc1_xyzm	52	5.64	24.58	15.11
bfcc2_xyzm	52	4.95	20.82	12.88
mfcc1_xyzm	52	4.90	18.96	11.93
bfcc1bfcc2_xyzm	104	5.33	22.09	13.71
bfcc1mfcc1_xyzm	104	5.52	20.00	12.76
bfcc2mfcc1_xyzm	104	5.02	18.23	11.63
bfcc1bfcc2mfcc1_xyzm	156	5.27	19.46	12.36

Table 8.8.: FMR and FNMR for different combinations of MFCC and BFCC (segment length 7.5 s, sampling rate 50 Hz).

Feature Vector	Feature Length	FMR	FNMR	HTER
mfcc1_x	13	3.83	40.68	22.26
mfcc1_y	13	3.92	47.80	25.86
mfcc1_z	13	4.90	54.78	29.84
mfcc1_m	13	3.87	40.18	22.02
mfcc1_xm	26	3.65	26.44	15.05
mfcc1_xyzm	52	4.90	18.96	11.93

Table 8.9.: FMR and FNMR for different axes and the magnitude vector (segment length 7.5 s, sampling rate 50 Hz).

Best Preprocessing Configuration

Table 8.10 states the results for the best feature vector and different applied sampling rates and segmentation lengths. All ten settings of the first session are used for training, setting 1 to 9 of the second session are used for testing. Sampling rate 25 Hz yields the worst results, but varying the sampling rate from 50 to 200 Hz has no significant influence to the error rates. As before, sampling rate 50 Hz is chosen for the following evaluations to reduce the computational complexity. Increasing the segment length from 2 s to 7.5 s is slightly decreasing the FNMR while keeping the FMR on the same level. This makes a segment length of 7.5 s most suitable.

MFCC_xyzm	25 Hz		50 Hz		100 Hz		200 Hz	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
2 s	3.21	34.75	3.85	30.48	3.99	30.47	4.06	30.46
3 s	3.45	28.41	3.99	26.17	4.04	26.24	4.09	26.65
4 s	3.53	25.90	4.22	22.89	4.17	23.34	4.32	23.89
5 s	3.66	24.47	4.33	21.70	4.13	22.33	4.20	22.73
6 s	3.92	23.93	4.46	21.68	4.32	21.13	4.44	21.23
7.5 s	4.13	22.19	4.90	18.96	4.80	18.74	4.79	19.55

Table 8.10.: FMR and FNMR for feature set MFCC_xyzm using various sampling rates and segment lengths.

Optimal Amount of Training Data

Figure 8.14 depicts the influence of the amount of training data. As before, the random sequences of setting 1 to 9 given in Table E.1 in Appendix E are applied. When increasing the number of settings which are used for training, the HTER is steadily decreasing. The best result with $\text{FMR} = 4.96\%$ and $\text{FNMR} = 18.91\%$ is obtained when using all nine settings of the first walk as training data². This corresponds to around four minutes walking. Adding more training data further reduces the error rates slightly. An FMR of 5.29% at an FNMR of 15.65% is obtained when setting 1 to 9 of walk 1 and 2 are used for training. Further increasing the training set by adding setting 1 to 9 of the third walk yields an FMR of 5.29% at an FNMR of 13.65%. To keep the training duration at an acceptable level for the user four to five minutes of training data are considered to be sufficient. Therefore, in the following sections the stated results are obtained when using setting 1 to 9 of the first walk at the first session for training, as before.

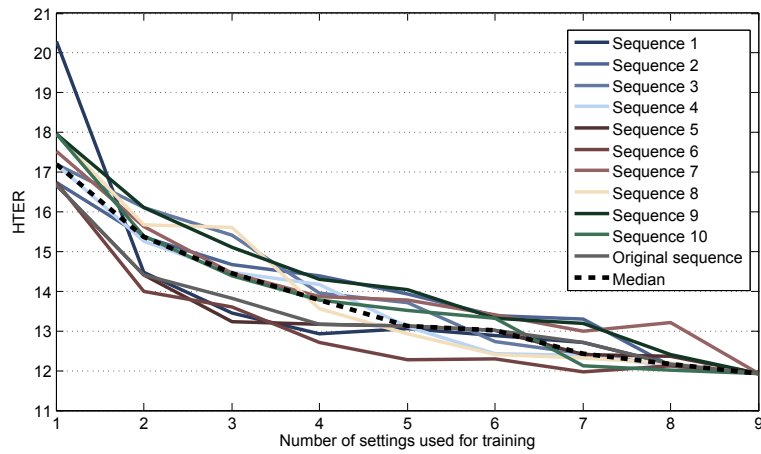


Figure 8.14.: HTER for different amounts of training data (sampling rate 50 Hz, segment length 7.5 s, feature set MFCC_xyzm).

Separate Results for Each Setting

Separate results of each setting are given in Figure 8.15 in terms of FMR and FNMR. The FMR is stable over all settings, but the sections containing stairs yield much higher FNMRs than the remaining sections.

Same-day Results

The influence of template aging can be analyzed by comparing the previously stated result ($\text{FMR} = 4.96\%$ and $\text{FNMR} = 18.91\%$) obtained when using test data of the second session (first walk, all nine settings) with the ones obtained when using test data of the first session (second walk, all nine settings). In both cases the models are trained using the first walk of the first session. The

²The slight variation to the previously stated results is due to the fact that setting 0 is not part of the random sequences.

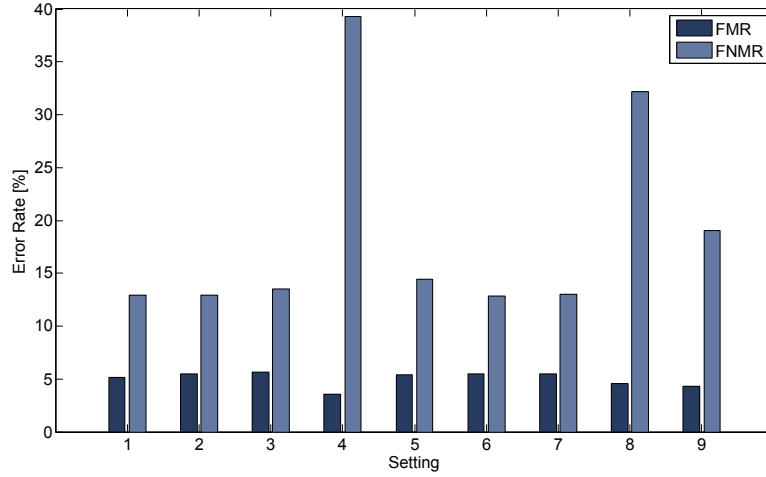


Figure 8.15.: FMR and FNMR at each setting for segment length 7.5 s, sampling rate 50 Hz, feature set MFCC_{xyzm}.

FMR stays on the same level (5.10%) and the FNMR is highly reduced to 3.34%. This shows that there is a high variability of the gait over time.

Subject-wise Results

A good classifier gives stable results for all subjects. To evaluate the k-NN algorithm against this property, separate results for each subject are calculated. FMR and FNMR for each subject are given in Figure 8.16. The results are similar to the ones of HMMs showing only a few outliers.

Voting

Although the FNMRs of the k-NN algorithm are lower than the ones of the other two classifiers, there is an unbalanced proportion between FNMR and FMR. This situation is improved by applying the voting algorithm. Figure 8.17 states the results for the so far used segment length 7.5 s and for 3 s, as this segment length delivers better results after voting. For segment length 7.5 s, the maximal group size $\#V$ is 30, for segment length 3 s $\#V$ can be increased to 100. Segments between one and 18 votes are required to vote for genuine.

From the DET-curve the EER of 6.7% can be determined for segment length 3 s. The closest FMR-FNMR-pair (FMR = 7.02% and FNMR = 6.25%) is obtained for $\#V = 70$ and $\#GV = 14$. Table 8.11 gives the HTERs for segment length 3 s. As for the SVM-results, the HTER is low when the ratio of $\#GV$ and $\#V$ is between 1:12 and 1:3 (marked gray). The lowest HTER of 5.31 (FMR = 8.49 and FNMR = 2.13) is obtained for $\#V = 100$ and $\#GV = 16$.

8.5. Cycle-based Method

The cycle extraction method is also evaluated on this database. The results are published in [110]. Cycles are extracted as described in Section 5.2.1 and the acceleration values of the cycles are

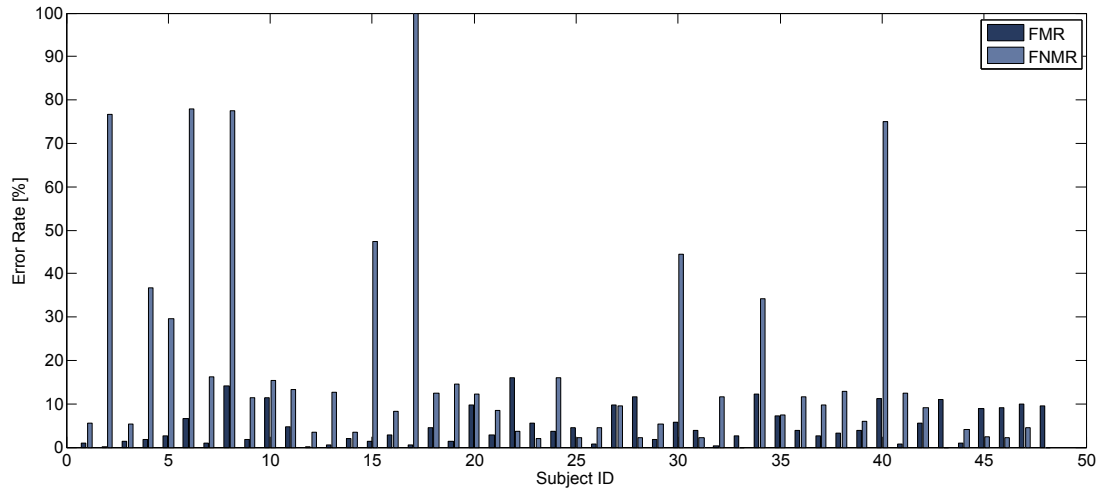


Figure 8.16.: FNMR and FMR for each of the 48 subjects when using the k-NN algorithm for classification.

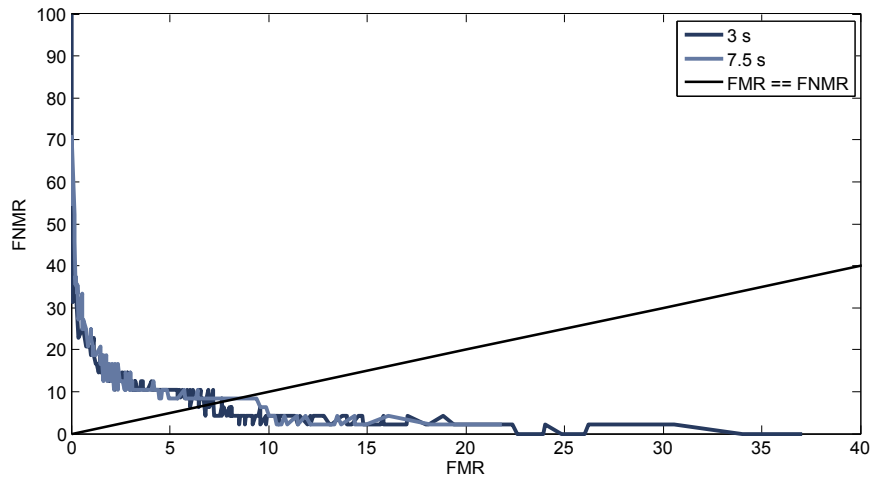


Figure 8.17.: DET-curve obtained by plotting the FMRs and FNMRs determined for different $\#GV$ and $\#V$ (feature set MFCC_xyzm, sampling rate 50 Hz, and two different segment lengths).

HTER	10	20	30	40	50	60	70	80	90	100
1	8.87	11.34	13.04	14.16	16.32	17.02	17.67	17.83	18.10	18.52
2	7.91	9.07	10.61	11.50	11.42	11.31	11.96	12.43	12.58	13.00
3	7.41	7.37	8.41	9.46	9.53	10.11	10.73	11.34	11.84	12.26
4	7.72	7.14	7.37	8.22	8.95	9.38	10.00	9.53	10.03	10.79
5	6.91	8.68	6.33	7.33	8.14	8.49	9.03	8.45	8.95	9.52
6	6.33	8.14	6.95	6.52	7.33	7.87	8.34	8.03	8.53	8.86
7	9.19	7.72	8.80	6.29	6.68	7.06	7.49	8.22	7.60	8.17
8	12.04	7.49	8.45	6.02	6.41	6.76	7.22	7.76	7.02	7.66
9	12.77	8.07	8.03	7.68	6.10	6.52	6.79	7.30	7.60	7.12
10	19.87	7.76	7.80	8.45	5.87	6.18	6.56	7.06	7.33	6.82
11	–	7.33	7.41	8.22	7.60	5.79	6.37	6.60	7.03	6.31
12	–	8.18	8.30	7.84	8.49	5.48	6.10	6.49	6.83	6.01
13	–	10.11	7.87	7.60	8.22	7.45	6.95	6.29	6.76	5.85
14	–	9.96	7.76	7.14	7.95	8.30	6.64	6.14	6.56	5.66
15	–	9.88	7.56	7.03	7.64	7.91	7.45	6.99	6.18	5.47
16	–	10.80	7.53	6.83	7.37	7.76	7.22	7.76	7.18	5.31
17	–	12.81	7.22	6.79	7.14	7.57	7.06	7.52	7.99	6.18

Table 8.11.: HTER obtained for #GV between 1 and 17 and #V between 10 and 100. Those values for which the ratio between #GV and #V is between 1:12 and 1:3 are marked gray.

compared using DTW and CRM (see Section 5.4.1). The distinctiveness of this evaluation is that it is completely performed on the smartphone. Two modules are integrated in MBASSy:

Majority Voting Module During calculation of the probe cycles the last step is *deletion of unusual cycles* and the remaining cycles are used as probe cycles. To calculate the reference cycles, during enrolment the *best cycle* is determined. In both cases, the cycles are not normalized in length. Comparison is done using DTW as distance function and applying majority voting: The distances of the reference cycle to all probe cycles are computed. If the distance between two cycles is below a pre-selected threshold this is called a match, otherwise a non-match. If at least 50% of the results are a match, the whole comparison is assumed to be a match and the subject is authenticated.

Cyclic Rotation Metric Module During enrolment and authentication, step *deletion of unusual cycles* is the last one and the remaining cycles are used as reference or probe, respectively. Cycles have to be normalized as CRM is used for comparison.

Depending on the module, one or several reference cycles were computed from data of setting 0 and stored on the phone during enrolment. The computation of the reference template(s) takes in average 3.1 seconds (minimum 2.1, maximum 4.9) for the majority voting module and 2.4 seconds (minimum 1.2, maximum 4.8) for the cyclic rotation metric module (see Table 8.13). The longer time for the majority voting module is due to the additional step which selects the best cycle.

The parameters chosen in the evaluation are determined empirically and are given in Table 8.12. The parameters depend on the selected sampling rate, because this directly influences the magnitudes of the entries of the salience vector.

Cycle Detection Step	Parameter Name	Evaluated Value
Interpolation	Sampling Rate S	150 Hz
Estimation of Cycle Length	Min. Peak Height Factor H_e	$0.8 * S = 120$
	Min. Peak Distance Factor D_e	$0.5 * S = 75$
Cycle Detection	Min. Peak Height Factor H_d	$0.7 * S = 105$
	Unusual Length Factor U_d	$0.2 * S = 30$
Length Normalization	Length L	original cycle length, 150
Omitting Unusual Cycles	Distance Factor D_o	$0.3 * S = 50$

Table 8.12.: The evaluated parameters defining the cycle extraction process.

To perform the evaluation under realistic conditions, a separate authentication application was written for the smartphone, which extracted the cycles, computed the dissimilarity distances, authentication results, and times used for cycle extraction and comparison and stored them in an internal table on the phone. It was tested beforehand that the times computed using this separate application are comparable to the ones obtained using the regular authentication process.

8.5.1. Evaluation

Using the previously described authentication application, probe cycles were extracted and compared to the reference data. For each subject and each module two reference templates exist, one for each session. An interesting point is to see the influence of the reference template and the influence of the time period between enrolment and authentication. Therefore, the data of the two sessions were separated. Each reference template was compared with the data from the same session and the other session (on a different day). Figure 8.18 and 8.19 show the DET-curves. The dark blue lines (R1Px) show the results when using the reference template recorded during the first session, the medium blue lines (R2Px) show the results obtained with the reference template of the second session. The dashed lines (RxP0) show the same-day results, the continuous line (RxP1) the different-day results and the dash-dotted-lines (RxP2) show the result when data of both days is used for testing. The EERs can be seen from the crossing with the diagonal. One can clearly see that the error rates increase when enrolment and probe data are not from the same day.

The light blue lines (R3Px) correspond to the results obtained when using the best reference template. This is determined by computing the FNMRs using the same-day probe data. For each subject that template is chosen which has the lower number of false non matches. This process would in a real scenario correspond to a training phase: Several templates are computed during enrolment. The one which gives the best performance (in terms of FNMR) during the same day is finally stored as the reference template. The continuous light blue line (R3P1) gives the most realistic results of all tests: A training phase ensures that a high-quality template is used as reference, and this reference will in general be compared with data which are not collected on the same day. This means that the CRM module yields an EER of 21.7% and the Majority Voting (MV) module of 28.0%.

Separate Results for Each Test Setting

The results were further analyzed separately for each section (see Figure 8.20). The given FNMRs are obtained at an FMR of approximately 10% for each module, while using the previously

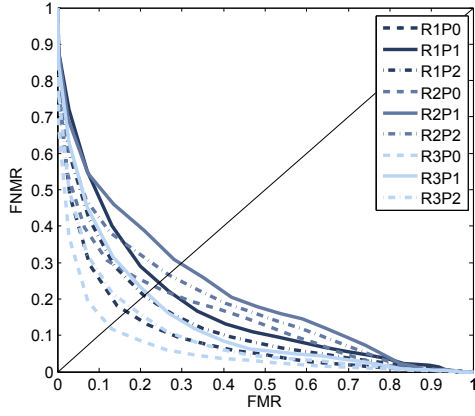


Figure 8.18.: DET-curves for CRM method using different reference and probe data.

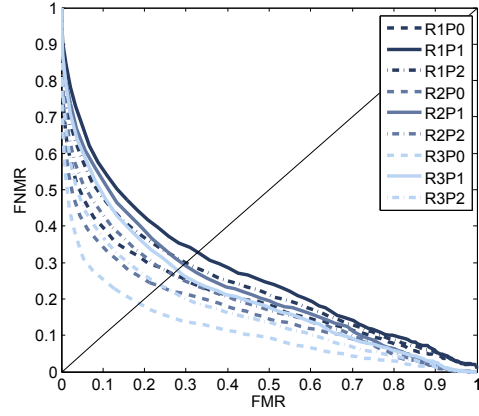


Figure 8.19.: DET-curves for majority voting method using different reference and probe data.

determined best reference template and probe data of a different day. One can see that for both modules the FNMRs are nearly the same for all sections, only section 4 and 8 show worse results, due to containing walking on stairs. This is conform to the results of the machine learning algorithms.

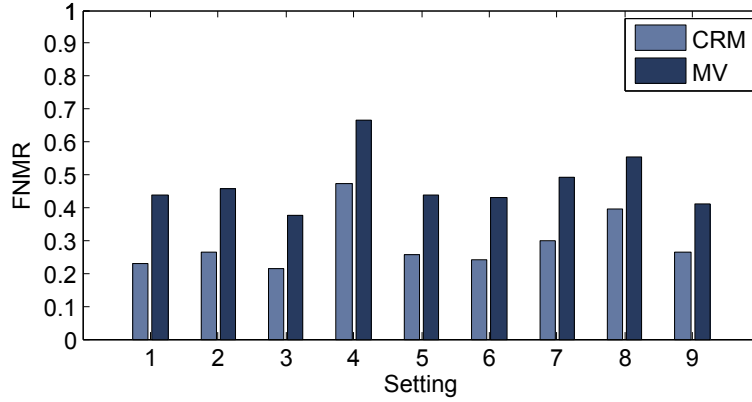


Figure 8.20.: FNMR separated by setting at an FMR of approximately 10% obtained for the cycle-based method.

Subject-wise Results

Figure 8.21 shows that the authentication results greatly depend on the subject. Some subjects are never recognized, whereas some are always. One reason for this is that some subjects (8, 13,

15, 17, and 27) wore different shoes during the two sessions. For subject 8 and 13, the influence of the different shoes is not noticeable, but it clearly is for the other three subjects.

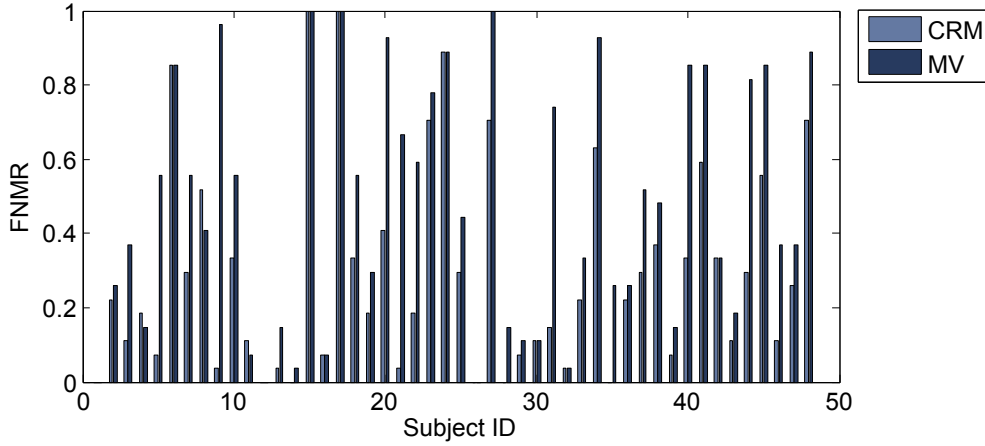


Figure 8.21.: FNMR separated by subject obtained by applying the cycle-based method (at an FMR of ca. 10%).

Computational Complexity

As one can see from the runtimes given in Table 8.13, extracting probe cycles from data of one section and comparing these cycles with the stored reference cycle(s), takes around 32 seconds for the CRM-module and about 27 seconds for the MV-module. The longer computation time for the CRM-module is due to the length normalization of the cycles and the expensive cyclic rotation metric. These durations are far too high for a real authentication application, as a user would have to wait around half a minute until the phone unlocks itself, which is much more than entering a PIN would take. This situation can be improved by performing a continuous authentication. This means that the module collects 30 seconds of data, directly starts the authentication, stores the authentication result and starts collecting data again for the next 30 seconds and so on. The achieved recognition rates and runtimes are good enough to implement this approach. When the user of the mobile phone starts the authentication, the last authentication result just needs to be obtained from memory, which could be done fast enough to be unnoticed by the user. With about 3 seconds for each module, the enrolment times are very good.

8.6. Summary and Conclusions

In this chapter, the different proposed gait recognition methods, using HMMs, SVMs, the k-NN algorithm, and the cycle extraction technique, were evaluated on a challenging database. Table 8.14 gives an overview of the best obtained EER for the four different methods. The biometric performances for the three approaches using machine learning algorithms for comparison are all on the same level. The results obtained with the cycle extraction method and using the raw cycles as features are much worse.

		CRM	MV
Extraction of reference cycle(s)	min	1179	2135
	max	4823	4880
	mean	2381	3067
Extraction of probe cycles	min	8758	7408
	max	65573	58021
	mean	26210	22843
Comparison	min	920	84
	max	107628	1864
	mean	5685	372

Table 8.13.: Mean times (in milliseconds) needed for cycle extraction and comparison.

	Feature Set	EER
HMM	BFCC2MFCC_xyzm	6.13
SVM	BFCC1MFCC_xm	6.38
k-NN	MFCC_xyzm	6.70
Cycle Extraction	raw data	21.7

Table 8.14.: Lowest obtained EER for each of the four evaluated algorithms (after voting).

The following six research questions are addressed in this chapter:

Research Question 1: Are machine learning algorithms more suitable than the formerly proposed gait recognition methods using gait cycles and template-based comparisons? Classification via machine learning algorithms and a template-based comparison using gait cycles are evaluated on the same database. For HMMs the best obtained EER is 6.13%, for SVMs an EER of 6.38% was obtained, and the k-NN algorithm yields an EER of 6.70%. The gait cycle method results in a much higher EER of 21.7%. Although there is potential to improve the gait cycle method, e.g. by replacing the applied majority voting by the more flexible quorum voting, the results show that the machine learning classification is the more promising approach.

Research Question 2: What are suitable features to describe subject specific gait? This chapter focused on feature sets consisting of different combination of MFCCs and BFCCs for different axes and axes-combinations. The best feature set for HMMs was BFCC2MFCC_xyzm, which was already identified in Chapter 7 to be the best feature set. For SVMs the best results were obtained using BFCC1MFCC_xm. Although the feature set identified in Chapter 7 was different (BFCC1BFCC2_xm), the combination of x-axis and magnitude vector m is outperforming the other axes. For the k-NN algorithm the situation is different. Here, best results are obtained when combining all three axes and the magnitude vector. The best feature set is MFCC_xyzm.

Research Question 3: How much training data are necessary when applying machine learning algorithms to accelerometer-based biometric gait recognition? The influence of the amount of training data is evaluated by increasing the number of settings used for training from 1 to 9 in ten different random sequences. The original sequence was used as

well and the median was analyzed. For HMMs the results for the different sequences varied much but eight settings (slightly more than three minutes) are considered to be sufficient to obtain well-trained models. SVMs required at least seven settings for training, corresponding to about three minutes. The influence of the applied sequence was not as big as for HMMs but outliers occurred, resulting in high error rates. The k-NN algorithm yielded the most stable results and a smooth decrease of the error rates up to using nine settings for training. The corresponding 3.7 minutes of walking data are considered as minimal amount of training data necessary. In all three cases the required amount of training data and corresponding enrolment effort is low enough for practical applications.

Research Question 4: Does the biometric performance decrease with aging of the reference data? For all algorithms same-day and cross-day results were presented. These show a high decrease of the biometric performance when training and testing data are not collected on the same day. This fact should be considered when gait recognition methods are evaluated to obtain more realistic results.

Research Question 6: How is the biometric performance of the proposed methods in a scenario test? The performance of the methods in a realistic scenario was the main concern of this chapter. Comparing the obtained error rates to the ones of the previous chapters, one can see no degradation of the error rates. In fact, the results are even improved. In addition, acceptable error rates can already be obtained using a lower training duration.

Research Question 7: Are current smartphones capable of performing the necessary feature extraction and classification steps with an acceptable computational complexity? In addition to the approach using the k-NN algorithm (see Section 7.4.2), the cycle extraction method was implemented as MBASSy application and run directly on the phone. The runtimes show that the phone is capable of performing the necessary steps in reasonable time. Implemented as continuous authentication module, authentication using the cycle extraction method can be performed without interfering the user.

The contribution of this thesis is the development and evaluation of new approaches for accelerometer-based biometric gait recognition. The aimed application scenario is the authentication on smartphones and the main research question to answer was: *Can accelerometer-based biometric gait recognition be used as authentication method on smartphones?* To answer this question, the developed approaches have been thoroughly evaluated on three different gait databases collected using two different off-the-shelf smartphones. The contributed new approaches veer away from the usually applied methods which are based on gait cycles and regular distance calculations. Instead of segmenting gait cycles from the accelerometer data, a segmentation based on fixed-length time intervals is proposed. The extracted features have not been used for biometric gait recognition before. For classification, machine learning algorithms are applied, namely HMMs, SVMs, and the k-NN algorithm. A scenario test was conducted to confirm the quality of the new approaches by a comparison with an advanced cycle-extraction method on a realistic database. Additionally, a benchmark to a published state-of-the-art method was performed. It was shown, that the contributed algorithms outperform this method.

9.1. Answers to Research Questions

Based on the evaluation results it is possible to answer the research questions given in the introduction:

Research Question 1: Are machine learning algorithms more suitable than the formerly proposed gait recognition methods using gait cycles and template-based comparisons? The conducted evaluations show that the recognition rates obtained from machine learning algorithms are better than those of cycle-based authentication. A first hint of this conclusion is already given in Chapter 6. The evaluations using HMMs and SVMs are performed on the same database as the evaluation of a cycle extraction method, published earlier [41]. Although the results are not directly comparable due to different partitions into training and testing data, the potential of HMMs and SVMs is shown by a decrease of EER by 50%.

In Chapter 8, SVMs, HMMs, the k-NN algorithm, and a newly proposed cycle extraction method are evaluated on a challenging data set. The best results obtained for machine learning algorithms are an EER of 6.13% for HMMs, an EER of 6.38% for SVMs, and an EER of 6.70% for the k-NN algorithm. The best EER for the cycle extraction method was 21.7%. Although the EER of the gait cycle method might be improved e.g. by replacing the used majority voting by the more flexible quorum voting or by using the RDTW distance, the results show that the classification using machine learning algorithms is the more promising approach.

In addition to a low overall EER, the stability of the results over all subjects is also important. The evaluations show, that HMMs and the k-NN algorithm give the most stable results and the error rates for the cycle extraction method vary highly for the different subjects.

Research Question 2: What are suitable features to describe subject specific gait? The evaluated statistical features like mean or standard deviation are not capable to describe the subject-specific gait sufficiently. Although MFCCs and BFCCs have been developed for speaker and speech recognition, it was shown, that they are very suitable for accelerometer-based biometric gait recognition. Replacing the Mel or Bark scale by a linear scale in the feature extraction process did degrade the results slightly. Using HMMs for classification of normal walk, combining BFCC2 and MFCC features from all three axes and magnitude vector (BFCC2MFCC_xyzm) is identified as best feature vector because it gives good results for both normal walk settings of database 2 and for database 3. For fast walk using only the combination of x-axis and magnitude vector has to be preferred and feature vector BFCC1_xm performed best.

For SVMs the combination of x-axis and magnitude vector gave best results for fast and normal walk. Feature set BFCC1BFCC2_xm gave best results for normal walk of database 2 and BFCC1MFCC_xm was best for database 3. Both feature set have also acceptable results each on the other database, hence both of them are suitable for classification using SVMs. For fast walk feature set BFCC2_xm yielded best results.

Evaluation of the k-NN algorithm on database 2 showed that feature vector BFCC2_xyzm is best for normal walk and on database 3, feature vector MFCC_xyzm outperformed the others. Each of them does also deliver good results on the other database, making them both suitable for classification using the k-NN algorithm. For fast walk BFCC1_xm should be chosen, affirming that the combination of x-axis and magnitude vector is the optimal choice for fast walk.

For cycle-based classification, the default approach is to use the raw accelerometer data. Extracting BFCC or MFCC features from the gait cycles and using HMMs for classification gives similar results to when fixed-length segments are used.

Research Question 3: How much training data is necessary when applying machine learning algorithms to accelerometer-based biometric gait recognition? The influence of the amount of training data was evaluated on two different databases. On each database, the required amount of training data lies between 3 and 5 minutes for the machine learning algorithms. Therefore, the walking duration during enrolment should take 5 minutes. Because enrolment is only performed seldom, this time is acceptable.

Research Question 4: Does the biometric performance decrease with aging of the reference data? In the evaluations using database 2 and 3, there is a clear temporal separation between training and testing data. The results are referred to as cross-day results, because they have been recorded on two different days. This is the realistic scenario, as one would not perform the

enrolment each day. These cross-day results are compared to the same-day results, where training and testing data stem from the same day. Depending on the method and database, the same-day results are only one third of the cross-day results. Hence, there is a high aging effect visible.

Research Question 5: How is the biometric performance when reference and probe are obtained from data collected with different walking velocities? Because database 2 contains data where the subjects were walking in normal velocity and in fast velocity, it was possible to evaluate the influence of different velocities on the recognition rates. For all three classifiers, the obtained EERs and HTERs were more than twice as high for the cross-velocity comparisons as for same-velocity comparisons. Interestingly, using data of both velocities for training, kept the biometric performance on the same level as when training and probe data were stemming from same single velocity. Therefore, either different models should be trained for different walking velocities or training data of different velocities should be used to train one model.

Research Question 6: How is the biometric performance of the proposed methods in a scenario test? HMMs, SVMs, and the k-NN algorithm are evaluated on database 2, containing only walking straight on flat floor, and on database 3 collected during a scenario test. Surprisingly, the error rates are not degraded. The lowest EER obtained for HMMs on normal walk of database 2 is 8.47% and on database 3 an EER of 6.13% is obtained using the same feature set. For SVMs the HTER/EER is also lower on database 3 (6.38 instead of 8.68) and the same holds for the k-NN algorithm. Here, the HTER is 6.70% instead of 7.64%. This shows that the proposed machine learning algorithms are capable to describe the subject-specific walk in a way that is robust enough to yield an acceptable performance in a realistic scenario.

Research Question 7: Are current smartphones capable of performing the necessary feature extraction and classification steps with an acceptable computational complexity? So far, two of the described methods have been implemented as MBASSy modules to be used for classification on the smartphone. For the cycle-based method it is shown in Section 8.5 that the time needed for computation of the reference cycle is very low with around 3 seconds¹. When probe cycles are extracted from around 25 seconds of data, the calculation and comparison to the reference cycle takes around 32 seconds. This would be a too long waiting time for the user, if the calculations would be started when an authentication result is required. But the duration is short enough to implement cycle extraction as a continuous authentication module, performing iteratively 40 seconds of data collection, followed by an authentication. In case an authentication result is required, it is possible to get it directly from MBASSy without any delay.

Out of the machine learning approaches the k-NN algorithm is implemented as MBASSy module. The best identified feature set and preprocessing configurations are used. Five minutes of walking data are collected during enrolment. Training of the classifier takes around 1.5 minutes² when the impostor set consists of twelve walks of each of 20 subjects. The authentication is based on the classification of 30 seconds acceleration data. Feature extraction and classification take around seven seconds. As before, the implementation as continuous module is possible, resulting in a fast and userfriendly authentication.

¹Evaluated on a Motorola Milestone.

²Evaluated on a Motorola Defy.

Combining the results of all stated research questions allows answering the main research question *Can accelerometer-based biometric gait recognition be used as authentication method on smartphones?* in the affirmative. The developed methods have low error rates and current smartphones are capable of performing the necessary training and testing in a short time. Accelerometer-based biometric gait recognition can be implemented as continuous authentication, meeting the demands of smartphone users for a fast authentication.

9.2. Benchmark of the Algorithms

Table 9.1 gives an overview over the most important properties of the evaluated algorithms. The length of the best feature vector lies between 52 and 104 elements. The authentication should be based on 0.25 to 3.2 minutes of walk data which are used for voting. The lowest EER/HTER for normal walk is between 7.46 and 9.63 on database 2 and between 6.13 and 21.7 for database 3. A walking duration of 5 minutes during training was identified to be best for all algorithms. The EER for the cycle extraction method is much higher than the one obtained from the machine learning algorithms. Based on the error rates, no clearly best machine learning algorithm can be identified. A disadvantage of SVMs was that outliers occurred training, which resulted in very high FNMRs. If this occurs during enrolment, the created classifier is unsuitable and the enrolment has to be repeated. To retain a high usability it would be necessary to detect this problematic data and dismiss it from the training procedure. k-NN gives the lowest HTER after voting, but has also the lowest error rate before voting. Hence, it gives already good results when classification is based on a short walk duration. Therefore, the k-NN algorithm is identified to be most suitable. The algorithm has been implemented as MBASSy module. It has been shown that the computational effort needed for feature extraction, training, and classification is low enough such that the algorithm can be run on the smartphone.

Algorithm	Length of best feature vector	Authentication based on ... minutes	Lowest EER/HTER on DB2	Lowest EER on DB3
HMM	104	2.5	8.47	6.13
SVM	52	2.5	8.68	6.38
k-NN	52	3.2	7.64	6.70
Cycle Extraction	~100	0.25	–	21.7

Table 9.1.: Comparison of the different evaluated approaches for normal walk.

9.3. Simulation Study

At the end of 2010, a simulation study was conducted by University of Kassel at CASED. The goal of this study was to get information about the evidentiary value of different prototypes in case of litigations. Beside digital watermarks and pseudonymized positioning data, accelerometer-based gait recognition was evaluated.

In cooperation between the technical developers and legal experts, four fictive legal case descriptions were developed for gait recognition. Although the application scenario described in this thesis focuses on authentication on smartphones, the technique can be used in various

applications. In the developed legal cases a photo camera with integrated accelerometer is the recording device. The idea in the different scenarios is that the camera is bound to the user by applying gait-recognition. In one legal case it is analyzed if the stored reference data from two cameras can be used to determine if they belong to the same user. In a further one, a photo should be used to proof that the owner of the camera has been abroad and is not be responsible for actions taken in his name in his home town. The cases are based in criminal law, social law, and civil law.

These case files, transcripts, and evidences have been given to the participating lawyers, which are acting for their fictive clients as lawyers, attorney, or representatives of the government. Based on documents from the lawyers, the participating judges instruct expert witnesses to give their opinion to the technologies. The expert witnesses did also participate at the simulated court situation, lasting two days.

Main result of the simulated court situation was that the three developed techniques are mainly secure for their purpose and field of application. Because they are not intended as documentary evidence, in most hearings they have just been used as evidentiary hint. In case the amount of money involved in the case was low, manipulations are unlikely because of the too high complexity. The error rates for accelerometer-based gait recognition of 5-10% assumed in the simulation were too high, especially for criminal law. Therefore, the rule *benefit of the doubt* had to be applied.

9.4. Future Work

One limitation of the conducted research is the fixed position of the phone which has to be in a pouch attached to the hip of the subject. Because the survey conducted in Section 2.4 showed that the most popular place to carry the mobile phone is the front trouser pocket, future research is necessary to allow for an authentication independent of the sensor position.

A further limitation is that the training of the stochastic classifiers requires feature vector instances of impostors which implies that a complete product would need to ship with a database with the feature instances. Although this is not impractical, from a privacy point of view it would be preferable to have pre-trained models that are only further trained with the instances of the genuine user during enrolment. Therefore incremental learning approaches, like [153] should be evaluated as well.

When biometric data is stored on an external server, several security and privacy risk occur. The operator of the system (controller of the data base) or an attacker might extract sensitive information from the data (e.g. health status). Because it is hard to learn a new gait, replacing a compromised template is not easily possible. In addition, a biometric template is unique for a specific subject. Therefore, cross-matching between different databases for different applications is possible, making it easy for an attacker to construct profiles of those data subjects enrolled in multiple applications. Due to the variability of biometric data, cryptographic encryption is not applicable. Therefore, the development of secure template protection methods for gait data is mandatory.

Gait recognition is only applicable in case the subject is walking. Therefore, a practical solution should incorporate activity recognition to identify the relevant data. In addition, classification of the walking speed is necessary. The evaluations showed that for different walking speeds, different feature vectors perform best. Furthermore, there is a high decrease in the biometric performance in case reference and probe data are stemming from data collected with different walk velocities.

Database 3 contained walking data collected during a realistic scenario. The achieved good

classification performances show the high robustness of the algorithms against walking around corners, or being distracted, e.g. by door sills. But there are several other covariates like undergrounds, shoes, trousers or backpacks which have to be considered. In addition, the recognition performance should be improved, e.g. by removing outliers from the training data to improve the reference data.

The influence of aging of the biometric templates is only evaluated by comparing same-day and cross-day results. To identify how frequently an enrolment should be performed, gait data needs to be collected at fixed time intervals over a longer time-period. This allows for an evaluation of the biometric performance depending on the time between enrolment and authentication (like in [88] for vision-based biometric gait recognition).

APPENDIX A

SURVEY

Survey: Security of Mobile Devices

The goal of this data collection is to gain information about the behavior of users of mobile phones and to get an impression about their security awareness. The results of this survey can be used for simulations and will serve as a basis for further research projects in this area.

DATA PROTECTION: Participation in this survey is voluntary and anonymous. All information will be kept confidential. According to these principles, analysis and publication of the results will occur completely anonymized.

The survey will take about four minutes. We kindly ask you to answer the questions. It is not necessary to answer each question. For specific questions multiple answers are possible. These are indicated by (*).

	1	2	3	4
I use my mobile phone (1 = privately, 4 = for business)				
For me, my mobile phone is (1 = very important, 4 = unimportant)				
Knowledge of your phone's features (1 = very good, 4 = no interest)				
Knowledge of IT-security (1 = very good, 4 = no interest)				

I carry my phone mainly		
<input type="checkbox"/> in my trousers' back pocket	<input type="checkbox"/> in my trousers' front pocket	<input type="checkbox"/> in my breast pocket
<input type="checkbox"/> in a pouch at the belt	<input type="checkbox"/> in my backpack	<input type="checkbox"/> in my hand bag
<input type="checkbox"/> in the pocket of my jacket	<input type="checkbox"/> further _____	
I use following additional functionality of my mobile phone (*)		
<input type="checkbox"/> SMS	<input type="checkbox"/> e-mail	<input type="checkbox"/> internet
<input type="checkbox"/> camera	<input type="checkbox"/> calendar	<input type="checkbox"/> further _____
Following data is stored on my mobile phone (*)		
<input type="checkbox"/> phone numbers	<input type="checkbox"/> addresses	<input type="checkbox"/> e-mails
<input type="checkbox"/> appointments	<input type="checkbox"/> passwords/PINs	<input type="checkbox"/> birthdays
<input type="checkbox"/> photos	<input type="checkbox"/> further _____	
When I use my mobile phone after a stand-by phase (*)		
<input type="checkbox"/> I have to unlock the key lock	<input type="checkbox"/> I have to enter a PIN	
<input type="checkbox"/> I can directly use it	<input type="checkbox"/> further _____	
If no PIN is necessary after a stand-by phase, why did you choose this setting?		
<input type="checkbox"/> is faster	<input type="checkbox"/> PIN is difficult to remember	<input type="checkbox"/> did not think about it
<input type="checkbox"/> my phone does not offer the possibility to enter a PIN after stand-by phase		
<input type="checkbox"/> further _____		
If your phone would offer authentication via biometrics instead of PIN, would you use this? If so, which biometric modalities would you use? (*)		
<input type="checkbox"/> no	<input type="checkbox"/> yes, face recognition	<input type="checkbox"/> yes, fingerprint recognition
<input type="checkbox"/> yes, speaker recognition	<input type="checkbox"/> yes, gait recognition	<input type="checkbox"/> not specified
My mobile phone		
<input type="checkbox"/> is always near me	<input type="checkbox"/> is often unattended	<input type="checkbox"/> is sometimes lent to sb. else
<input type="checkbox"/> further _____		
My mobile phone has already been stolen/lost		
<input type="checkbox"/> no	<input type="checkbox"/> yes, _____times	
If somebody stole your phone, where did this happen? (*)		
<input type="checkbox"/> public transport	<input type="checkbox"/> city center	<input type="checkbox"/> office
<input type="checkbox"/> in a crowd of people	<input type="checkbox"/> further _____	
I am		
<input type="checkbox"/> male	<input type="checkbox"/> female	<input type="checkbox"/> not specified
Please select your age group		
<input type="checkbox"/> below 18	<input type="checkbox"/> 18-23	<input type="checkbox"/> 24-30
<input type="checkbox"/> 31-40	<input type="checkbox"/> 41-50	<input type="checkbox"/> 51-60
<input type="checkbox"/> above 60	<input type="checkbox"/> not specified	

APPENDIX B

HMM RESULTS FOR DATABASE 2

Sampling rate	50			100			200		
Segment length	3 s	5 s	7.5 s	3 s	5 s	7.5 s	3 s	5 s	7.5 s
BFCC1_xyzm	20.93	19.85	19.84	20.93	19.92	19.83	20.36	20.22	20.14
BFCC2_xyzm	17.51	18.08	17.89	17.79	18.39	18.25	17.90	18.41	18.16
MFCC_xyzm	17.58	16.96	16.63	17.63	17.19	16.75	17.59	16.95	16.45
BFCC1BFCC2_xyzm	19.54	19.13	19.46	19.39	19.24	18.94	19.08	19.08	19.19
BFCC1MFCC_xyzm	19.31	18.94	18.85	19.10	18.89	18.80	18.85	18.76	18.85
BFCC2MFCC_xyzm	17.14	17.28	17.27	17.28	17.56	17.50	17.21	17.49	17.39
BFCC1BFCC2MFCC_xyzm	18.56	18.76	18.83	18.60	18.70	18.57	18.39	18.61	18.60

Table B.1.: HMM results for training with data from walk 1-12 of setting 1 for different sampling rates and segment lengths.

Sampling rate	50			100			200		
Segment length	3 s	5 s	7.5 s	3 s	5 s	7.5 s	3 s	5 s	7.5 s
BFCC1_xm	14.86	14.37	13.76	15.47	15.01	14.47	15.54	15.17	14.87
BFCC2_xxm	15.94	15.34	15.32	16.13	15.71	16.18	16.31	16.06	16.68
MFCC_xxm	16.62	16.5	15.75	16.83	17.13	16.51	16.99	17.2	16.95
BFCC1BFCC2_xxm	14.88	14.41	13.85	15.27	14.87	14.58	15.42	15.28	15
BFCC1MFCC_xxm	14.83	14.47	13.83	15.32	15.19	14.46	15.45	15.29	14.76
BFCC2MFCC_xxm	15.51	15.37	14.72	15.81	15.9	15.5	16.06	16.25	16.2
BFCC1BFCC2MFCC_xxm	14.84	14.4	13.73	15.18	15.11	14.54	15.4	15.23	14.92

Table B.2.: HMM results for training with data from walk 1-12 of setting 2 for different sampling rates and segment lengths.

Sampling rate	50			100			200		
Segment length	3 s	5 s	7.5 s	3 s	5 s	7.5 s	3 s	5 s	7.5 s
BFCC1_xyzm	16.29	14.99	14.60	16.85	15.04	14.46	17.34	15.89	15.30
BFCC2_xyzm	15.38	14.03	13.91	15.65	14.22	13.75	15.96	14.55	13.80
MFCC_xyzm	16.22	14.94	15.01	16.39	15.05	15.30	16.73	15.24	15.38
BFCC1BFCC2_xyzm	15.60	13.90	13.40	15.85	14.22	13.33	16.29	14.59	13.93
BFCC1MFCC_xyzm	15.62	14.14	14.07	15.98	14.41	14.03	16.41	14.91	14.74
BFCC2MFCC_xyzm	15.42	14.20	13.98	15.58	14.27	14.27	15.92	14.46	14.38
BFCC1BFCC2MFCC_xyzm	15.33	13.86	13.45	15.68	14.08	13.55	16.05	14.41	14.08

Table B.3.: HMM results for training with data from walk 1-12 of setting 3 for different sampling rates and segment lengths.

Sequence ID	Walk IDs											
1	4	1	6	5	10	2	7	3	8	9	12	11
2	7	11	4	12	8	6	2	9	5	1	10	3
3	5	8	6	10	4	2	1	11	7	9	12	3
4	5	11	9	7	10	2	12	4	1	6	8	3
5	1	2	4	11	12	10	3	5	6	9	7	8
6	7	1	4	9	8	10	2	6	5	3	12	11
7	11	6	9	4	2	3	7	5	12	8	1	10
8	12	3	2	8	1	4	9	5	7	10	6	11
9	3	11	5	6	9	12	2	1	4	8	10	7
10	3	12	9	4	5	8	6	11	1	2	10	7

Table B.4.: Ten randomly created sequences of the available 12 normal walks used to analyse the influence of the amount of training data.

Sequence ID	Walk IDs															
1	16	15	8	10	14	4	1	5	9	12	3	7	11	6	13	2
2	3	16	7	9	2	10	6	11	13	5	8	4	15	1	14	12
3	3	5	9	4	16	12	13	15	10	7	14	1	8	2	6	11
4	4	5	16	11	8	15	7	12	3	10	14	13	6	2	1	9
5	1	5	4	2	11	12	9	16	15	10	6	14	13	3	7	8
6	10	9	14	2	13	4	11	1	5	16	7	6	15	3	12	8
7	3	6	2	16	8	12	10	5	15	4	11	9	13	7	14	1
8	4	16	3	10	8	15	12	7	1	6	14	13	9	5	11	2
9	10	13	3	8	1	4	15	9	11	14	12	16	2	7	6	5
10	12	14	2	7	6	15	3	16	11	13	4	5	9	1	8	10

Table B.5.: Ten randomly created sequences of the 16 available fast walks used to analyse the influence of the amount of training data.

Training sequence	Number of raining walks											
	1	2	3	4	5	6	7	8	9	10	11	12
1	25.69	18.10	18.14	18.94	17.46	17.16	16.84	17.13	17.29	17.19	17.03	17.28
2	21.49	20.18	19.35	19.15	18.81	18.91	17.93	17.88	17.97	17.23	17.24	17.28
3	24.71	21.01	21.12	19.76	18.65	17.98	17.46	17.68	17.05	17.05	17.24	17.28
4	24.71	22.77	21.09	19.37	19.05	17.84	18.02	17.35	16.73	16.96	17.24	17.28
5	20.46	18.55	17.38	16.29	16.61	16.78	16.91	17.24	17.36	17.44	17.08	17.28
6	21.49	16.72	16.58	16.37	16.94	16.33	16.17	16.60	16.95	17.19	17.03	17.28
7	24.17	22.31	21.80	19.83	18.10	18.26	17.65	17.79	17.83	18.12	17.27	17.28
8	21.30	18.66	19.55	19.82	17.95	17.12	17.15	17.44	16.81	16.77	17.03	17.28
9	21.99	19.20	19.65	20.08	19.73	19.77	19.37	18.07	17.46	17.86	17.80	17.28
10	21.99	18.66	18.68	17.89	17.98	18.62	18.95	19.29	18.03	17.86	17.80	17.28
original	20.46	18.55	19.20	18.31	18.94	19.06	18.21	18.40	17.91	17.19	17.17	17.28
median	21.99	18.66	19.35	19.15	18.10	17.98	17.65	17.68	17.36	17.19	17.24	17.28

Table B.6.: EERs obtained for different numbers of training walks from session 1 (training and testing data of setting 1, feature set BFCC2MFCC_xyzm). The used training sequences are given in Table B.4.

Training sequence	Number of training walks											
	1	2	3	4	5	6	7	8	9	10	11	12
1	20.95	17.02	16.68	16.43	15.14	15.13	15.40	14.73	14.41	14.09	14.25	14.20
2	21.35	18.11	15.97	15.41	15.21	15.09	14.96	14.48	14.73	14.72	14.50	14.20
3	20.12	16.41	16.24	14.91	14.79	14.71	14.98	14.85	14.96	14.43	14.50	14.20
4	20.12	18.42	17.07	16.23	15.10	14.71	14.75	14.75	14.92	14.80	14.50	14.20
5	20.96	19.54	17.03	16.16	15.77	15.24	14.86	14.80	14.74	14.50	14.57	14.20
6	21.35	19.39	16.48	16.06	14.87	14.46	14.37	14.36	14.48	14.09	14.25	14.20
7	19.70	18.17	16.47	15.66	15.07	14.39	14.39	14.62	14.77	14.37	14.29	14.20
8	19.09	16.49	16.12	14.27	14.84	14.8	14.50	14.50	14.34	14.23	14.25	14.20
9	18.99	15.90	16.50	16.35	15.22	15.02	14.75	14.87	14.93	14.34	14.28	14.20
10	18.99	16.49	15.02	14.87	15.48	14.28	14.27	14.30	14.28	14.34	14.28	14.20
original	20.96	19.54	16.52	16.27	15.77	15.48	15.06	14.5	14.18	14.09	14.15	14.20
median	20.12	18.11	16.48	16.06	15.14	14.80	14.75	14.62	14.73	14.34	14.28	14.20

Table B.7.: EERs obtained for different amounts of training walks from session 1 (training and testing data of setting 3, feature set BFCC2MFCC_xyzm). The used training sequences are given in Table B.4.

B. HMM Results for Database 2

Training walks	1	2	3	4	5	6	7	8
1	23.26	18.83	17.09	16.57	16.60	16.04	15.20	15.08
2	24.07	18.45	16.38	15.91	16.24	15.42	15.35	15.53
3	24.07	20.23	17.12	16.73	16.27	16.08	14.83	15.03
4	22.46	18.53	16.52	17.74	16.97	16.58	15.75	15.77
5	25.78	15.89	15.51	16.09	15.85	14.89	14.54	15.00
6	23.00	15.48	15.37	14.58	13.78	13.98	14.29	13.60
7	24.07	18.27	18.48	16.49	16.83	16.36	16.23	16.37
8	22.27	15.39	16.30	16.73	16.51	16.62	16.49	15.78
9	19.75	17.99	16.39	15.50	15.31	13.78	13.50	13.81
10	21.45	17.62	15.63	14.66	14.32	14.42	14.60	15.12
original	25.78	19.28	18.84	18.13	16.82	16.04	15.38	15.30
median	23.26	18.27	16.39	16.49	16.27	16.04	15.20	15.12

Training walks	9	10	11	12	13	14	15	16
1	14.87	14.96	15.18	14.82	14.91	14.84	14.48	14.48
2	14.86	15.04	15.05	15.01	15.08	14.53	14.51	14.48
3	15.42	15.10	14.98	14.46	14.50	14.38	14.47	14.48
4	15.82	16.06	15.72	15.35	15.11	15.05	14.53	14.48
5	15.01	15.40	15.32	15.10	14.70	14.66	14.45	14.48
6	13.64	14.56	14.46	14.40	14.36	14.49	14.45	14.48
7	16.19	16.07	16.27	16.09	15.26	15.04	14.99	14.48
8	15.00	14.82	14.79	14.39	14.31	14.38	14.48	14.48
9	13.87	14.12	14.71	14.78	14.39	14.29	14.38	14.48
10	15.26	14.80	14.75	14.74	14.75	14.28	14.38	14.48
original	14.64	14.39	14.55	14.37	14.01	13.97	13.91	14.48
median	15.00	14.96	14.98	14.78	14.70	14.49	14.47	14.48

Table B.8.: EERs obtained for different amounts of training walks from session 1 (training and testing using data of setting 2) for feature set BFCC1_xm. The used training sequences are given in Table B.5.

APPENDIX C

SVM RESULTS FOR DATABASE 2

C. SVM Results for Database 2

Segment length	2 s			3 s			4 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
BFCC1_x	3.61	51.96	27.78	3.10	51.16	27.13	2.58	49.47	26.02
BFCC2_x	3.43	53.45	28.44	2.98	47.01	25.00	2.15	44.33	23.24
MFCC1_x	2.81	52.35	27.58	2.49	49.03	25.76	2.91	45.18	24.04
BFCC1_xm	2.63	44.66	23.65	2.11	45.19	23.65	1.50	41.57	21.53
BFCC2_xm	2.71	43.90	23.30	2.23	39.62	20.92	2.00	37.98	19.99
MFCC1_xm	2.98	46.63	24.81	2.06	44.46	23.26	1.68	43.40	22.54
BFCC1_xyzm	1.02	52.37	26.69	1.23	49.67	25.45	1.14	47.70	24.42
BFCC2_xyzm	1.07	52.10	26.58	1.20	45.03	23.11	0.89	46.28	23.59
MFCC1_xyzm	1.62	49.15	25.39	1.03	48.64	24.83	0.77	48.36	24.57
BFCC2bin10_xyzm	1.35	51.42	26.38	0.80	48.56	24.68	0.69	48.94	24.82
BFCC1bin10_xyzm	1.42	51.19	26.30	0.87	51.88	26.38	0.76	51.35	26.05
BFCC1BFCC2_xyzm	1.28	49.69	25.48	0.77	49.94	25.36	0.58	50.18	25.38
allFeat_x	2.20	49.86	26.03	1.53	46.45	23.99	2.12	40.81	21.46
allFeat_xm	1.87	44.07	22.97	1.35	42.72	22.03	1.14	43.45	22.29
BFCC1MFCC_xm	1.97	43.75	22.86	1.37	43.45	22.41	1.90	39.33	20.61
BFCC2MFCC_xm	2.48	43.85	23.16	1.96	42.68	22.32	1.53	41.72	21.62
BFCC1BFCC2_xm	2.02	45.15	23.59	1.54	43.90	22.72	2.07	39.60	20.84
BFCC1BFCC2MFCC_xm	2.20	43.19	22.70	1.80	41.67	21.74	1.59	41.65	21.62
BFCC1BFCC2_x	3.12	48.06	25.59	2.33	43.86	23.09	2.31	42.41	22.36
BFCC1BFCC2_y	3.57	65.90	34.74	3.02	65.16	34.09	2.49	66.89	34.69
BFCC1BFCC2_z	3.28	71.25	37.27	3.31	70.47	36.89	2.64	68.37	35.51
BFCC1BFCC2_m	3.49	56.70	30.09	3.33	53.38	28.36	2.83	50.49	26.66

Segment length	5 s			6 s			7.5 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
BFCC1_x	2.38	50.56	26.47	3.30	46.28	24.79	3.30	44.92	24.11
BFCC2_x	2.13	46.02	24.08	1.85	46.63	24.24	1.42	47.27	24.35
MFCC1_x	2.91	43.74	23.32	2.91	43.75	23.33	2.60	42.58	22.59
BFCC1_xm	1.66	42.72	22.19	1.27	42.90	22.09	1.09	42.46	21.78
BFCC2_xm	1.87	39.39	20.63	1.47	39.38	20.43	1.24	38.56	19.90
MFCC1_xm	1.52	41.23	21.37	1.22	43.83	22.53	1.20	42.35	21.77
BFCC1_xyzm	0.75	49.46	25.11	0.67	48.66	24.66	0.67	49.81	25.24
BFCC2_xyzm	0.87	45.77	23.32	0.77	43.69	22.23	0.58	44.24	22.41
MFCC1_xyzm	0.71	48.28	24.49	0.54	49.10	24.82	0.46	49.02	24.74
BFCC2bin10_xyzm	0.58	50.16	25.37	0.48	48.14	24.31	0.35	50.04	25.19
BFCC1bin10_xyzm	0.51	52.09	26.30	0.44	51.75	26.09	0.28	53.52	26.90
BFCC1BFCC2_xyzm	0.48	51.12	25.80	0.39	51.37	25.88	0.26	53.41	26.84
allFeat_x	2.23	41.93	22.08	2.08	40.08	21.08	1.94	41.52	21.73
allFeat_xm	0.98	44.02	22.50	0.77	42.79	21.78	0.63	42.88	21.75
BFCC1MFCC_xm	2.04	39.41	20.73	1.49	39.65	20.57	1.27	38.90	20.09
BFCC2MFCC_xm	1.67	39.53	20.60	1.40	42.26	21.83	1.21	40.04	20.62
BFCC1BFCC2_xm	0.92	46.32	23.62	2.00	39.38	20.69	1.77	38.48	20.13
BFCC1BFCC2MFCC_xm	1.43	40.97	21.20	1.10	42.09	21.60	0.99	40.27	20.63
BFCC1BFCC2_x	2.17	42.30	22.23	2.01	41.94	21.98	1.70	42.58	22.14
BFCC1BFCC2_y	2.47	69.43	35.95	2.02	68.27	35.14	1.78	70.38	36.08
BFCC1BFCC2_z	2.33	70.71	36.52	2.11	68.53	35.32	1.79	68.90	35.34
BFCC1BFCC2_m	2.31	49.74	26.03	2.10	51.72	26.91	1.90	49.81	25.85

Table C.1.: Cross-day results using sampling rate 25 Hz and different segment lengths for setting 1.

Segment length	2 s			3 s			4 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
BFCC1_x	3.03	51.30	27.17	2.69	50.31	26.50	2.80	49.07	25.94
BFCC2_x	2.75	52.88	27.81	2.45	47.38	24.92	2.19	43.82	23.01
MFCC1_x	2.67	51.24	26.95	2.44	50.36	26.40	2.10	47.14	24.62
BFCC1_xm	2.52	41.24	21.88	1.72	43.11	22.42	1.29	42.05	21.67
BFCC2_xm	2.19	42.54	22.37	1.77	39.96	20.87	1.37	38.93	20.15
MFCC1_xm	2.08	45.56	23.82	1.64	44.97	23.31	1.19	42.45	21.82
BFCC1_xyzm	0.68	53.35	27.02	0.57	50.55	25.56	0.55	49.38	24.97
BFCC2_xyzm	0.64	54.30	27.47	0.97	46.82	23.90	0.68	45.59	23.13
MFCC1_xyzm	1.15	52.03	26.59	0.96	49.73	25.35	0.51	48.34	24.43
BFCC2bin10_xyzm	0.86	52.53	26.70	0.68	50.00	25.34	0.54	49.82	25.18
BFCC1bin10_xyzm	0.98	52.00	26.49	0.57	51.09	25.83	0.44	49.78	25.11
BFCC1BFCC2_xyzm	0.81	50.45	25.63	0.45	50.52	25.49	0.30	50.22	25.26
allFeatures_x	1.88	48.33	25.10	1.30	48.80	25.05	2.14	39.55	20.85
allFeatures_xm	1.24	44.15	22.69	0.84	43.73	22.29	0.58	41.87	21.23
BFCC1BFCC2_xm	1.71	41.76	21.73	0.95	46.97	23.96	0.77	45.31	23.04
BFCC1MFCC_xm	1.60	44.29	22.95	1.16	45.25	23.21	1.45	38.33	19.89
BFCC2MFCC_xm	1.55	45.27	23.41	1.13	44.71	22.92	1.78	38.44	20.11
BFCC1BFCC2MFCC_xm	1.83	41.53	21.68	1.49	40.57	21.03	1.27	39.22	20.25
BFCC1BFCC2_x	3.09	49.82	26.45	2.26	43.49	22.87	2.01	42.34	22.18
BFCC1BFCC2_y	3.74	65.13	34.43	2.97	65.44	34.21	2.55	68.02	35.28
BFCC1BFCC2_z	2.48	69.22	35.85	1.99	68.44	35.21	1.72	68.36	35.04
BFCC1BFCC2_m	3.17	51.84	27.50	2.47	51.43	26.95	1.85	51.17	26.51

Segment length	5 s			6 s			7.5 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
BFCC1_x	3.51	43.01	23.26	3.53	44.68	24.10	3.63	42.58	23.11
BFCC2_x	2.85	38.80	20.83	2.41	39.13	20.77	2.05	42.12	22.09
MFCC1_x	2.69	44.77	23.73	2.83	44.77	23.80	2.71	46.23	24.47
BFCC1_xm	1.03	42.91	21.97	0.84	42.97	21.90	0.71	43.00	21.86
BFCC2_xm	1.15	39.37	20.26	1.02	38.52	19.77	0.71	39.00	19.86
MFCC1_xm	1.14	42.54	21.84	1.06	44.19	22.62	0.91	44.48	22.70
BFCC1_xyzm	0.43	49.45	24.94	0.45	48.46	24.45	0.32	49.01	24.67
BFCC2_xyzm	0.64	47.90	24.27	0.57	47.59	24.08	0.52	47.37	23.95
MFCC1_xyzm	0.36	50.18	25.27	0.37	50.70	25.53	0.42	49.73	25.08
BFCC2bin10_xyzm	0.49	50.90	25.70	0.44	51.45	25.95	0.34	52.74	26.54
BFCC1bin10_xyzm	0.41	50.48	25.45	0.36	51.22	25.79	0.25	52.66	26.46
BFCC1BFCC2_xyzm	0.26	51.61	25.94	0.22	52.82	26.52	0.19	54.15	27.17
allFeatures_x	2.13	40.40	21.27	1.89	39.80	20.85	1.87	40.49	21.18
allFeatures_xm	0.48	41.43	20.96	0.40	41.51	20.95	0.33	42.58	21.46
BFCC1BFCC2_xm	1.17	39.86	20.52	1.20	36.54	18.87	1.04	39.08	20.06
BFCC1MFCC_xm	1.61	38.71	20.16	1.12	39.68	20.40	1.30	36.23	18.77
BFCC2MFCC_xm	1.29	37.65	19.47	1.49	38.14	19.81	0.89	36.64	18.77
BFCC1BFCC2MFCC_xm	1.05	40.56	20.81	0.96	40.03	20.49	0.76	39.23	20.00
BFCC1BFCC2_x	2.07	42.11	22.09	1.77	42.82	22.29	1.64	44.87	23.26
BFCC1BFCC2_y	2.20	69.41	35.81	1.91	69.83	35.87	1.56	72.59	37.07
BFCC1BFCC2_z	1.80	69.92	35.86	1.62	69.74	35.68	1.54	70.19	35.86
BFCC1BFCC2_m	1.72	50.94	26.33	1.74	53.14	27.44	1.56	51.41	26.48

Table C.2.: Cross-day results using sampling rate 50 Hz and different segment lengths for setting 1.

C. SVM Results for Database 2

Segment length	2 s			3 s			4 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
BFCC1_x	3.40	51.17	27.28	2.80	49.81	26.31	2.85	47.38	25.12
BFCC2_x	3.20	52.79	27.99	2.60	47.80	25.20	2.80	37.30	20.05
MFCC1_x	2.65	51.93	27.29	2.12	50.15	26.14	2.83	44.21	23.52
BFCC1_xm	2.48	43.94	23.21	1.77	46.05	23.91	1.45	43.74	22.59
BFCC2_xm	2.08	43.41	22.74	1.63	38.74	20.19	1.24	38.80	20.02
MFCC1_xm	2.22	44.47	23.35	1.51	44.92	23.22	1.30	42.72	22.01
BFCC1_xyzm	1.31	51.91	26.61	0.97	52.60	26.79	0.81	53.58	27.20
BFCC2_xyzm	0.98	50.93	25.96	0.79	47.03	23.91	0.58	46.89	23.74
MFCC1_xyzm	0.92	51.30	26.11	0.87	50.10	25.49	0.49	48.11	24.30
BFCC2bin10_xyzm	0.84	52.93	26.88	0.62	50.59	25.61	0.49	50.50	25.49
BFCC1bin10_xyzm	1.13	53.85	27.49	0.80	54.06	27.43	0.59	53.40	27.00
BFCC1BFCC2_xyzm	0.86	51.26	26.06	0.42	51.93	26.18	0.28	53.76	27.02
allFeatures_x	1.83	48.65	25.24	2.25	43.35	22.80	2.06	40.31	21.18
allFeatures_xm	1.35	43.98	22.67	0.98	43.91	22.45	0.57	41.64	21.10
BFCC1BFCC2_xm	1.77	42.73	22.25	0.99	47.89	24.44	1.42	37.67	19.54
BFCC1MFCC_xm	1.56	45.89	23.72	1.10	45.68	23.39	1.58	38.96	20.27
BFCC2MFCC_xm	1.63	44.57	23.10	1.20	44.44	22.82	1.73	37.98	19.85
BFCC1BFCC2MFCC_xm	1.85	41.17	21.51	1.50	40.19	20.85	1.28	40.26	20.77
BFCC1BFCC2_x	3.13	48.88	26.00	2.47	42.94	22.71	2.07	40.97	21.52
BFCC1BFCC2_y	3.88	64.56	34.22	3.17	65.98	34.58	2.68	68.86	35.77
BFCC1BFCC2_z	2.46	70.29	36.37	2.02	67.61	34.82	1.72	69.54	35.63
BFCC1BFCC2_m	2.59	53.81	28.20	2.46	50.54	26.50	2.06	49.63	25.84

Segment length	5 s			6 s			7.5 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
BFCC1_x	2.28	48.50	25.39	3.56	44.07	23.81	3.47	42.99	23.23
BFCC2_x	2.81	39.12	20.97	2.48	38.44	20.46	2.49	41.20	21.85
MFCC1_x	2.43	43.84	23.14	2.98	43.84	23.41	2.70	45.01	23.86
BFCC1_xm	1.25	45.11	23.18	1.18	44.74	22.96	1.04	45.62	23.33
BFCC2_xm	1.11	38.77	19.94	0.99	39.19	20.09	0.71	40.05	20.38
MFCC1_xm	1.11	44.08	22.60	1.08	43.75	22.42	1.71	37.31	19.51
BFCC1_xyzm	0.62	51.60	26.11	0.67	52.82	26.74	0.62	51.03	25.83
BFCC2_xyzm	0.59	48.00	24.30	0.52	48.14	24.33	0.50	47.83	24.17
MFCC1_xyzm	0.36	50.14	25.25	0.41	51.37	25.89	0.37	52.32	26.35
BFCC2bin10_xyzm	0.47	52.58	26.53	0.43	52.79	26.61	0.34	53.89	27.12
BFCC1bin10_xyzm	0.49	53.45	26.97	0.48	55.98	28.23	0.42	54.80	27.61
BFCC1BFCC2_xyzm	0.28	54.42	27.35	0.25	55.84	28.05	0.21	56.86	28.54
allFeatures_x	2.09	41.85	21.97	1.95	38.79	20.37	1.88	41.27	21.58
allFeatures_xm	0.43	42.72	21.58	0.35	42.68	21.52	0.30	43.41	21.86
BFCC1BFCC2_xm	1.28	39.59	20.44	1.32	36.52	18.92	1.11	39.56	20.34
BFCC1MFCC_xm	1.47	38.72	20.10	0.93	39.60	20.26	1.18	36.24	18.71
BFCC2MFCC_xm	1.26	37.78	19.52	1.30	38.38	19.84	1.04	36.36	18.70
BFCC1BFCC2MFCC_xm	0.91	40.91	20.91	0.90	40.21	20.55	0.78	40.17	20.48
BFCC1BFCC2_x	2.16	42.48	22.32	1.87	43.52	22.70	1.67	45.39	23.53
BFCC1BFCC2_y	2.45	68.44	35.45	1.69	71.99	36.84	1.68	73.17	37.42
BFCC1BFCC2_z	1.72	70.77	36.24	1.56	71.82	36.69	1.57	71.61	36.59
BFCC1BFCC2_m	1.85	51.90	26.88	1.73	53.52	27.62	1.28	54.99	28.14

Table C.3.: Cross-day results using sampling rate 100 Hz and different segment lengths for setting 1.

Segment length	2 s			3 s			4 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
BFCC1 _x	3.37	49.97	26.67	2.54	49.46	26.00	2.90	48.92	25.91
BFCC2 _x	2.91	50.44	26.68	2.48	46.38	24.43	2.81	36.58	19.70
MFCC1 _x	2.76	51.28	27.02	2.38	49.73	26.06	2.97	43.15	23.06
BFCC1 _{xm}	2.27	46.04	24.15	1.82	45.98	23.90	1.54	45.88	23.71
BFCC2 _{xm}	2.11	42.71	22.41	1.84	39.42	20.63	1.18	38.68	19.93
MFCC1 _{xm}	2.12	45.89	24.01	1.52	45.40	23.46	1.26	43.07	22.17
BFCC1 _{xyzm}	1.12	52.57	26.84	0.82	53.42	27.12	0.84	54.06	27.45
BFCC2 _{xyzm}	1.06	49.56	25.31	0.77	48.09	24.43	0.54	46.65	23.59
MFCC1 _{xyzm}	0.83	51.87	26.35	0.81	49.75	25.28	0.40	48.27	24.33
BFCC2bin10 _{xyzm}	0.82	52.43	26.62	0.60	51.57	26.09	0.43	50.66	25.54
BFCC1bin10 _{xyzm}	1.14	54.22	27.68	0.68	54.32	27.50	0.64	54.06	27.35
BFCC1BFCC2 _{xyzm}	0.82	51.92	26.37	0.49	53.11	26.80	0.28	54.74	27.51
allFeatures _x	1.95	47.22	24.58	2.16	43.39	22.78	2.20	40.56	21.38
allFeatures _{xm}	1.16	44.30	22.73	0.89	44.60	22.75	0.57	42.13	21.35
BFCC1BFCC2 _{xm}	1.76	43.80	22.78	1.02	48.11	24.57	1.42	37.64	19.53
BFCC1MFCC _{xm}	1.59	47.31	24.45	1.10	45.61	23.36	1.24	40.45	20.85
BFCC2MFCC _{xm}	1.61	45.11	23.36	1.25	44.97	23.11	1.61	37.79	19.70
BFCC1BFCC2MFCC _{xm}	1.85	42.29	22.07	1.56	40.53	21.05	1.21	40.14	20.67
BFCC1BFCC2 _x	2.94	48.54	25.74	2.52	43.68	23.10	2.34	40.05	21.19
BFCC1BFCC2 _y	3.64	65.86	34.75	3.56	65.80	34.68	3.09	67.81	35.45
BFCC1BFCC2 _z	2.58	69.14	35.86	2.38	67.79	35.09	1.77	69.29	35.53
BFCC1BFCC2 _m	2.73	54.51	28.62	1.77	52.85	27.31	1.72	52.30	27.01

Segment length	5 s			6 s			7.5 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
BFCC1 _x	3.31	45.34	24.33	3.99	44.99	24.49	3.84	43.79	23.82
BFCC2 _x	2.82	38.67	20.75	2.43	38.02	20.22	2.47	40.55	21.51
MFCC1 _x	2.58	42.99	22.79	3.12	42.09	22.60	2.78	44.36	23.57
BFCC1 _{xm}	1.18	46.58	23.88	1.26	46.01	23.63	0.82	46.88	23.85
BFCC2 _{xm}	1.13	38.81	19.97	0.95	39.21	20.08	1.71	32.51	17.11
MFCC1 _{xm}	0.95	45.53	23.24	1.73	40.58	21.15	2.78	44.36	23.57
BFCC1 _{xyzm}	0.79	52.43	26.61	0.83	52.80	26.82	0.74	51.37	26.06
BFCC2 _{xyzm}	0.57	47.69	24.13	0.56	47.81	24.18	0.49	47.64	24.07
MFCC1 _{xyzm}	0.33	50.29	25.31	0.40	51.38	25.89	0.34	53.28	26.81
BFCC2bin10 _{xyzm}	0.43	52.20	26.32	0.42	54.11	27.26	0.35	55.11	27.73
BFCC1bin10 _{xyzm}	0.57	54.71	27.64	0.52	56.40	28.46	0.46	56.33	28.40
BFCC1BFCC2 _{xyzm}	0.27	54.26	27.27	0.27	55.91	28.09	0.21	56.71	28.46
allFeatures _x	2.03	40.95	21.49	1.86	39.01	20.44	1.85	40.93	21.39
allFeatures _{xm}	0.45	43.39	21.92	0.42	43.28	21.85	0.29	43.98	22.14
BFCC1BFCC2 _{xm}	1.25	39.96	20.61	1.11	37.50	19.30	1.04	39.48	20.26
BFCC1MFCC _{xm}	1.48	38.98	20.23	1.13	40.40	20.76	1.12	37.58	19.35
BFCC2MFCC _{xm}	1.18	38.15	19.67	1.40	37.70	19.55	0.93	37.58	19.26
BFCC1BFCC2MFCC _{xm}	0.91	41.18	21.05	1.01	40.52	20.77	0.76	40.59	20.68
BFCC1BFCC2 _x	2.16	42.78	22.47	1.87	42.38	22.12	1.86	45.43	23.65
BFCC1BFCC2 _y	2.66	70.25	36.45	1.95	71.39	36.67	1.77	73.82	37.79
BFCC1BFCC2 _z	1.75	69.92	35.84	1.56	72.12	36.84	1.60	72.07	36.83
BFCC1BFCC2 _m	1.83	52.43	27.13	1.77	55.97	28.87	1.31	55.34	28.32

Table C.4.: Cross-day results using sampling rate 200 Hz and different segment lengths for setting 1.

APPENDIX D

K-NN RESULTS FOR DATABASE 2

Segment length	3 s			5 s			7.5 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
MFCC_x	3.15	48.15	25.65	3.26	44.86	24.06	3.40	45.81	24.61
BFCC1_x	3.96	45.08	24.52	3.97	41.39	22.68	4.05	41.51	22.78
BFCC2_x	3.55	45.83	24.69	3.60	40.75	22.17	3.88	41.06	22.47
BFCC1MFCC_x	3.65	43.95	23.80	3.62	40.99	22.31	3.72	39.54	21.63
BFCC2MFCC_x	3.52	45.69	24.61	3.50	42.19	22.85	3.71	41.44	22.57
BFCC1BFCC2_x	3.75	41.72	22.73	3.79	37.74	20.77	3.80	37.02	20.41
BFCC1BFCC2MFCC_x	3.68	43.45	23.57	3.78	39.53	21.65	3.73	38.58	21.16
MFCC_xyzm	3.70	27.98	15.84	3.57	25.76	14.66	3.66	26.18	14.92
BFCC1_xyzm	3.93	32.04	17.98	4.19	28.72	16.45	4.29	28.58	16.44
BFCC2_xyzm	3.78	26.75	15.26	3.67	24.72	14.20	3.96	23.59	13.77
BFCC1MFCC_xyzm	3.72	27.57	15.65	3.79	25.36	14.57	3.82	25.15	14.49
BFCC2MFCC_xyzm	3.76	26.14	14.95	3.66	23.83	13.75	3.76	24.09	13.92
BFCC1BFCC2_xyzm	3.89	28.68	16.28	3.81	25.83	14.82	4.01	24.66	14.34
BFCC1BFCC2MFCC_xyzm	3.78	27.03	15.40	3.65	24.94	14.29	3.75	24.01	13.88
BFCC1_xm	4.13	31.23	17.68	4.24	29.02	16.63	4.10	27.25	15.67
BFCC2_xm	4.44	30.11	17.28	4.55	25.80	15.18	4.66	25.23	14.94
MFCC_xm	4.17	34.22	19.19	4.32	30.08	17.20	4.15	28.50	16.33
BFCC1MFCC_xm	4.22	29.67	16.95	4.13	26.37	15.25	3.96	24.20	14.08
BFCC2MFCC_xm	4.34	30.09	17.21	4.43	26.65	15.54	4.49	25.57	15.03
BFCC1MFCC_xm	4.22	29.67	16.95	4.13	26.37	15.25	3.96	24.20	14.08
BFCC1BFCC2_xm	4.38	28.37	16.38	4.34	25.12	14.73	4.30	23.97	14.13
BFCC1BFCC2MFCC_xm	4.30	28.53	16.42	4.25	25.05	14.65	4.17	23.86	14.01
DPS-0.4	4.30	25.92	15.11	4.33	24.11	14.22	4.35	24.16	14.25
DPS-0.3	4.31	26.02	15.16	4.34	24.18	14.26	4.36	24.16	14.26
DPS-0.2	4.30	25.95	15.13	4.36	24.21	14.29	4.35	24.01	14.18
DPS-0.1	4.29	26.14	15.21	4.35	23.90	14.12	4.32	23.44	13.88
DPS0	4.20	25.43	14.82	4.30	23.17	13.74	4.21	22.83	13.52
DPS0.1	4.02	25.84	14.93	4.10	23.64	13.87	4.03	22.26	13.15
DPS0.2	3.82	27.45	15.64	4.00	24.58	14.29	3.99	23.36	13.67
DPS0.3	3.78	30.44	17.11	4.04	28.41	16.22	3.99	26.26	15.12
DPS0.4	4.16	35.75	19.95	4.43	34.43	19.43	4.41	33.49	18.95
DPS0.5	3.80	43.34	23.57	4.20	41.22	22.71	4.39	39.95	22.17
DPS0.6	3.86	51.77	27.81	4.25	50.36	27.31	4.28	49.58	26.93
DPS0.7	5.16	63.11	34.14	5.26	62.21	33.74	5.48	61.42	33.45

Table D.1.: Cross-day results for the k-NN algorithm for setting 1 using sampling rate 50 Hz and different segment lengths.

Segment length	3 s			5 s			7.5 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
MFCC_x	3.19	47.55	25.37	3.28	44.92	24.10	3.49	45.69	24.59
BFCC1_x	4.00	44.44	24.22	3.94	42.76	23.35	4.21	41.27	22.74
BFCC2_x	3.59	45.45	24.52	3.71	41.66	22.68	3.94	40.36	22.15
BFCC1MFCC_x	3.67	43.72	23.69	3.74	41.61	22.68	3.89	40.78	22.33
BFCC2MFCC_x	3.51	45.34	24.43	3.54	41.66	22.60	3.69	41.54	22.61
BFCC1BFCC2_x	3.77	41.28	22.53	3.90	39.07	21.49	3.93	37.46	20.70
BFCC1BFCC2MFCC_x	3.73	43.15	23.44	3.86	40.30	22.08	3.82	39.29	21.56
MFCC_xyzm	3.70	28.05	15.87	3.64	25.92	14.78	3.75	26.22	14.98
BFCC1_xyzm	4.23	34.43	19.33	4.30	31.56	17.93	4.36	30.75	17.56
BFCC2_xyzm	3.95	27.15	15.55	3.78	24.46	14.12	3.97	22.22	13.09
BFCC1MFCC_xyzm	3.80	29.54	16.67	3.96	27.04	15.50	4.09	26.52	15.31
BFCC2MFCC_xyzm	3.84	26.01	14.92	3.82	23.47	13.65	3.93	23.17	13.55
BFCC1BFCC2_xyzm	4.14	29.26	16.70	4.08	26.67	15.38	4.24	24.77	14.51
BFCC1BFCC2MFCC_xyzm	3.89	27.36	15.63	3.87	25.14	14.50	4.01	23.63	13.82
BFCC1_xm	4.39	34.07	19.23	4.50	33.25	18.87	4.55	32.24	18.39
BFCC2_xm	4.22	30.18	17.20	4.49	26.74	15.62	4.54	25.15	14.85
MFCC_xm	4.13	34.03	19.08	4.25	30.85	17.55	4.21	28.85	16.53
BFCC1MFCC_xm	4.16	31.45	17.81	4.15	29.63	16.89	3.99	27.55	15.77
BFCC2MFCC_xm	4.29	30.22	17.25	4.33	27.77	16.05	4.52	25.76	15.14
BFCC1MFCC_xm	4.16	31.45	17.81	4.15	29.63	16.89	3.99	27.55	15.77
BFCC1BFCC2_xm	4.39	29.37	16.88	4.37	27.56	15.97	4.49	25.95	15.22
BFCC1BFCC2MFCC_xm	4.26	29.58	16.92	4.18	27.77	15.98	4.27	25.95	15.11
DPS-0.4	4.50	26.78	15.64	4.47	25.28	14.87	4.54	25.42	14.98
DPS-0.3	4.51	26.72	15.62	4.46	25.33	14.90	4.55	25.38	14.97
DPS-0.2	4.52	26.74	15.63	4.45	25.19	14.82	4.58	25.30	14.94
DPS-0.1	4.43	27.21	15.82	4.51	25.35	14.93	4.55	24.66	14.60
DPS0	4.35	25.70	15.03	4.36	24.18	14.27	4.43	23.51	13.97
DPS0.1	4.13	26.68	15.41	4.22	25.16	14.69	4.24	23.44	13.84
DPS0.2	4.01	28.12	16.07	4.20	25.75	14.98	4.18	23.67	13.93
DPS0.3	3.95	31.27	17.61	4.21	29.09	16.65	4.18	27.97	16.08
DPS0.4	4.25	36.19	20.22	4.47	34.66	19.56	4.49	33.57	19.03
DPS0.5	3.88	42.83	23.36	4.10	41.40	22.75	4.34	40.09	22.22
DPS0.6	3.94	51.64	27.79	4.16	50.66	27.41	4.28	49.81	27.05
DPS0.7	5.11	63.10	34.11	5.20	61.77	33.49	5.48	61.51	33.50

Table D.2.: Cross-day results for the k-NN algorithm for setting 1 using sampling rate 100 Hz and different segment lengths.

Segment length	3 s			5 s			7.5 s		
	FMR	FNMR	HTER	FMR	FNMR	HTER	FMR	FNMR	HTER
MFCC_x	3.23	47.74	25.48	3.39	45.43	24.41	3.46	45.35	24.41
BFCC1_x	4.04	45.12	24.58	4.18	44.82	24.50	4.59	43.86	24.23
BFCC2_x	3.63	45.52	24.57	3.78	41.61	22.69	4.05	39.33	21.69
BFCC1MFCC_x	3.70	44.96	24.33	3.80	42.66	23.23	4.02	41.39	22.70
BFCC2MFCC_x	3.55	45.48	24.51	3.59	41.86	22.73	3.87	40.85	22.36
BFCC1BFCC2_x	3.83	41.39	22.61	4.00	39.14	21.57	4.21	37.77	20.99
BFCC1BFCC2MFCC_x	3.71	43.50	23.61	3.88	41.02	22.45	4.03	39.18	21.60
MFCC_xyzm	3.67	27.50	15.59	3.67	25.69	14.68	3.75	25.65	14.70
BFCC1_xyzm	4.31	33.71	19.01	4.44	31.79	18.12	4.40	31.17	17.79
BFCC2_xyzm	4.15	27.02	15.59	3.88	24.47	14.17	4.11	22.26	13.18
BFCC1MFCC_xyzm	3.77	28.87	16.32	4.11	26.63	15.37	4.22	25.72	14.97
BFCC2MFCC_xyzm	3.94	25.81	14.88	3.90	23.10	13.50	4.00	23.09	13.55
BFCC1BFCC2_xyzm	4.29	29.21	16.75	4.27	26.16	15.21	4.45	24.31	14.38
BFCC1BFCC2MFCC_xyzm	3.95	26.98	15.47	4.05	24.35	14.20	4.21	23.40	13.80
BFCC1_xm	4.32	35.55	19.93	4.50	34.19	19.34	4.57	33.35	18.96
BFCC2_xm	4.33	30.50	17.42	4.53	27.21	15.87	4.55	25.46	15.00
MFCC_xm	4.10	34.34	19.22	4.30	31.06	17.68	4.07	29.42	16.75
BFCC1MFCC_xm	4.15	32.25	18.20	4.24	30.03	17.14	4.20	28.39	16.29
BFCC2MFCC_xm	4.37	30.47	17.42	4.31	28.29	16.30	4.36	26.64	15.50
BFCC1MFCC_xm	4.15	32.25	18.20	4.24	30.03	17.14	4.20	28.39	16.29
BFCC1BFCC2_xm	4.43	30.20	17.31	4.43	27.94	16.18	4.44	26.68	15.56
BFCC1BFCC2MFCC_xm	4.28	30.61	17.44	4.29	28.34	16.31	4.30	26.68	15.49
DPS-0.4	4.54	26.60	15.57	4.54	25.50	15.02	4.64	24.89	14.76
DPS-0.3	4.52	26.61	15.57	4.57	25.38	14.98	4.62	24.96	14.79
DPS-0.2	4.52	26.60	15.56	4.55	25.38	14.97	4.63	24.96	14.80
DPS-0.1	4.45	26.86	15.66	4.52	25.64	15.08	4.75	25.34	15.05
DPS0	4.34	25.56	14.95	4.48	24.25	14.37	4.68	23.32	14.00
DPS0.1	4.05	26.81	15.43	4.31	25.64	14.97	4.51	24.24	14.37
DPS0.2	4.06	28.35	16.21	4.28	26.02	15.15	4.32	24.01	14.16
DPS0.3	4.11	30.78	17.45	4.36	28.74	16.55	4.45	27.36	15.91
DPS0.4	4.26	35.83	20.04	4.50	34.47	19.48	4.53	33.80	19.16
DPS0.5	3.87	42.66	23.27	4.15	41.18	22.67	4.34	40.89	22.61
DPS0.6	3.85	52.32	28.08	4.11	50.93	27.52	4.26	49.85	27.06
DPS0.7	5.13	62.81	33.97	5.24	61.33	33.28	5.53	61.09	33.31

Table D.3.: Cross-day results for the k-NN algorithm for setting 1 using sampling rate 200 Hz and different segment lengths.

APPENDIX E

HMM RESULTS FOR DATABASE 3

Sequence ID	Setting IDs								
1	4	1	6	5	2	7	3	8	9
2	7	4	8	6	2	9	5	1	3
3	5	8	6	4	2	1	7	9	3
4	5	9	7	2	4	1	6	8	3
5	1	2	4	3	5	6	9	7	8
6	7	1	4	9	8	2	6	5	3
7	6	9	4	2	3	7	5	8	1
8	3	2	8	1	4	9	5	7	6
9	3	5	6	9	2	1	4	8	7
10	3	9	4	5	8	6	1	2	7

Table E.1.: Ten randomly created sequences of the available nine settings used to analyse the influence of the amount of training data.

Segment length	2 s	3 s	4 s	5 s	6 s	7.5 s
BFCC1_xyzm	19.63	18.68	18.39	18.01	17.94	17.37
BFCC2_xyzm	18.62	17.58	16.79	16.66	16.15	15.93
MFCC_xyzm	18.43	16.67	16.07	15.55	15.37	14.80
BFCC1BFCC2_xyzm	18.91	17.89	17.42	17.38	16.97	16.22
BFCC1MFCC_xyzm	18.40	17.25	16.94	16.59	16.22	15.61
BFCC2MFCC_xyzm	18.00	16.87	16.32	15.86	15.59	15.04
BFCC1BFCC2MFCC_xyzm	18.26	17.26	16.93	16.41	16.09	15.50
BFCC1_x	23.64	22.13	22.10	21.58	21.87	21.79
BFCC2_x	24.52	22.61	21.79	21.27	21.26	20.39
MFCC_x	23.64	22.13	22.10	21.58	21.87	21.79
BFCC1BFCC2_x	23.21	21.53	21.10	20.83	21.04	20.64
BFCC1MFCC_x	22.44	20.83	20.56	20.02	20.14	19.54
BFCC2MFCC_x	22.98	21.09	20.34	19.67	19.45	18.62
BFCC1BFCC2MFCC_x	22.45	20.75	20.27	19.77	20.03	19.58
BFCC1_y	25.11	22.98	22.36	21.68	20.96	20.51
BFCC2_y	25.46	23.20	22.19	21.27	21.13	20.59
MFCC_y	23.21	20.49	19.44	18.79	18.83	18.01
BFCC1BFCC2_y	24.44	22.58	21.64	21.19	20.73	20.01
BFCC1MFCC_y	23.28	20.80	19.62	19.01	18.83	17.94
BFCC2MFCC_y	23.23	20.87	20.11	19.38	19.10	18.28
BFCC1BFCC2MFCC_y	23.36	21.04	20.16	19.60	19.01	18.37
BFCC1_z	30.16	28.62	27.65	26.95	27.08	26.62
BFCC2_z	28.20	26.66	25.91	24.99	24.63	24.29
MFCC_z	28.16	26.03	25.27	24.49	24.32	23.88
BFCC1BFCC2_z	28.66	27.09	26.01	25.39	25.24	24.75
BFCC1MFCC_z	28.43	26.42	25.72	24.84	24.97	24.24
BFCC2MFCC_z	27.62	25.90	25.14	24.22	24.01	23.63
BFCC1BFCC2MFCC_z	27.92	26.32	25.52	24.43	24.46	23.95
BFCC1_m	25.98	23.61	22.52	21.90	21.08	20.64
BFCC2_m	24.58	22.38	20.84	19.99	19.48	18.89
MFCC_m	24.97	22.14	20.84	19.80	19.39	18.56
BFCC1BFCC2_m	24.52	22.27	21.00	20.47	19.65	18.88
BFCC1MFCC_m	24.48	21.93	21.00	20.20	19.54	18.75
BFCC2MFCC_m	24.11	21.65	20.49	19.38	19.08	18.25
BFCC1BFCC2MFCC_m	24.09	21.60	20.59	19.77	19.41	18.40
BFCC1_xm	20.55	18.87	18.42	18.49	18.41	17.57
BFCC2_xm	20.20	18.33	17.75	17.12	17.10	15.90
MFCC_xm	20.40	18.56	17.72	16.91	16.80	16.00
BFCC1BFCC2_xm	19.52	17.98	17.88	17.45	17.51	16.72
BFCC1MFCC_xm	19.56	17.93	17.55	17.16	17.17	16.35
BFCC2MFCC_xm	19.50	17.78	17.19	16.58	16.35	15.40
BFCC1BFCC2MFCC_xm	19.17	17.75	17.20	16.90	16.97	15.85

Table E.2.: Results for all evaluated feature vectors and segment lengths obtained using sampling rate 25.

Segment length	2 s	3 s	4 s	5 s	6 s	7.5 s
BFCC1_xyzm	18,83	17,85	17,61	17,24	17,03	16,44
BFCC2_xyzm	17.74	16.92	16.55	16.03	16.26	15.66
MFCC_xyzm	17.80	16.31	15.54	15.19	15.07	14.99
BFCC1BFCC2_xyzm	17.93	17.07	16.85	16.48	16.43	15.77
BFCC1MFCC_xyzm	17.68	16.57	16.39	16.02	16.04	15.43
BFCC2MFCC_xyzm	17.31	16.11	15.63	15.23	15.17	14.88
BFCC1BFCC2MFCC_xyzm	17.42	16.55	16.27	15.79	15.89	15.25
BFCC1_x	22.42	21.83	21.53	21.60	21.59	21.78
BFCC2_x	23.91	22.22	21.68	21.28	21.52	21.43
MFCC_x	22.42	21.83	21.53	21.60	21.59	21.78
BFCC1BFCC2_x	22.36	21.20	20.94	20.85	21.20	21.14
BFCC1MFCC_x	21.49	20.29	20.50	20.07	19.93	19.89
BFCC2MFCC_x	22.01	20.40	20.35	19.75	19.70	19.25
BFCC1BFCC2MFCC_x	21.57	20.12	20.40	19.92	20.03	19.79
BFCC1_y	24.70	23.11	22.05	21.33	20.45	20.04
BFCC2_y	25.09	23.09	22.34	22.04	21.71	20.59
MFCC_y	22.83	20.52	19.35	19.09	18.17	17.92
BFCC1BFCC2_y	24.50	22.47	21.65	21.02	20.71	19.95
BFCC1MFCC_y	22.87	20.58	19.47	18.85	18.45	17.98
BFCC2MFCC_y	22.86	20.92	19.95	19.46	19.17	18.17
BFCC1BFCC2MFCC_y	23.07	21.10	19.92	19.49	19.08	18.24
BFCC1_z	29.91	28.61	27.63	27.22	26.81	26.61
BFCC2_z	28.10	26.82	25.83	25.55	25.25	24.74
MFCC_z	28.20	26.53	26.08	25.57	25.66	25.08
BFCC1BFCC2_z	28.82	27.40	26.22	25.84	25.12	25.15
BFCC1MFCC_z	28.47	26.83	25.93	25.48	25.03	24.72
BFCC2MFCC_z	27.72	26.46	25.61	25.10	24.73	24.29
BFCC1BFCC2MFCC_z	28.16	26.70	25.72	25.19	24.70	24.60
BFCC1_m	22.93	20.86	19.63	18.95	18.75	18.13
BFCC2_m	23.03	20.66	19.59	18.51	18.31	18.03
MFCC_m	22.40	20.11	19.15	18.16	17.76	17.27
BFCC1BFCC2_m	22.32	20.18	18.73	17.71	17.45	17.07
BFCC1MFCC_m	21.82	19.71	18.53	17.60	17.12	16.77
BFCC2MFCC_m	22.12	19.91	18.65	17.37	17.12	16.82
BFCC1BFCC2MFCC_m	21.67	19.60	18.26	17.30	16.87	16.52
BFCC1_xm	19.15	17.95	17.46	17.24	17.14	16.91
BFCC2_xm	19.03	17.54	16.79	16.29	16.25	15.73
MFCC_xm	19.25	17.77	16.83	15.77	15.81	15.55
BFCC1BFCC2_xm	18.50	17.29	16.80	16.32	16.38	16.17
BFCC1MFCC_xm	18.58	17.09	16.52	16.09	16.21	16.27
BFCC2MFCC_xm	18.63	17.00	16.46	15.69	15.81	15.26
BFCC1BFCC2MFCC_xm	18.29	16.82	16.34	15.78	15.91	15.98

Table E.3.: Results for all evaluated feature vectors and segment lengths obtained using sampling rate 50.

Segment length	2 s	3 s	4 s	5 s	6 s	7.5 s
BFCC1_xyzm	19.11	18.22	17.79	17.71	17.52	17.11
BFCC2_xyzm	17.96	17.13	16.35	16.09	16.02	15.51
MFCC_xyzm	17.93	16.49	15.35	14.89	15.01	14.85
BFCC1BFCC2_xyzm	18.17	17.27	16.99	16.60	16.55	16.16
BFCC1MFCC_xyzm	17.91	16.84	16.25	15.97	16.23	16.03
BFCC2MFCC_xyzm	17.46	16.21	15.36	15.11	15.08	14.86
BFCC1BFCC2MFCC_xyzm	17.52	16.63	16.07	15.78	15.89	15.52
BFCC1_x	22.90	22.08	22.10	21.99	21.68	22.26
BFCC2_x	23.95	22.20	21.79	21.71	21.67	21.29
MFCC_x	22.90	22.08	22.10	21.99	21.68	22.26
BFCC1BFCC2_x	22.63	21.60	21.17	21.09	21.48	21.08
BFCC1MFCC_x	21.62	20.43	20.29	20.35	20.16	19.98
BFCC2MFCC_x	22.04	20.36	20.01	19.85	19.68	19.04
BFCC1BFCC2MFCC_x	21.65	20.32	20.50	20.34	20.07	20.13
BFCC1_y	24.99	22.94	22.21	21.68	20.56	20.16
BFCC2_y	25.22	23.57	22.47	21.94	21.51	21.32
MFCC_y	22.82	20.48	19.16	19.01	18.31	18.25
BFCC1BFCC2_y	24.59	22.74	21.46	21.00	20.36	20.09
BFCC1MFCC_y	22.69	20.69	19.40	18.85	18.16	17.70
BFCC2MFCC_y	22.74	20.73	19.72	19.19	18.83	18.32
BFCC1BFCC2MFCC_y	22.90	21.10	19.91	19.14	18.49	18.20
BFCC1_z	29.91	28.45	27.53	27.25	26.60	26.83
BFCC2_z	28.23	27.21	26.16	25.79	25.53	25.35
MFCC_z	28.13	26.70	26.07	25.58	25.76	25.26
BFCC1BFCC2_z	28.82	27.46	26.37	25.93	25.37	25.51
BFCC1MFCC_z	28.38	26.93	25.86	25.53	25.22	24.87
BFCC2MFCC_z	27.83	26.55	25.71	25.26	25.10	24.80
BFCC1BFCC2MFCC_z	28.12	26.85	25.81	25.24	24.97	24.69
BFCC1_m	24.73	22.61	21.43	20.61	20.52	19.81
BFCC2_m	23.03	20.65	19.46	18.37	18.28	18.12
MFCC_m	22.73	20.28	19.40	18.10	18.19	17.87
BFCC1BFCC2_m	23.03	20.97	19.52	18.56	18.39	18.04
BFCC1MFCC_m	22.73	20.84	19.50	18.50	18.26	17.80
BFCC2MFCC_m	22.12	19.98	18.70	17.66	17.56	17.20
BFCC1BFCC2MFCC_m	22.47	20.33	19.10	18.04	17.80	17.25
BFCC1_xm	20.10	19.08	18.17	18.15	18.16	17.84
BFCC2_xm	19.22	17.55	16.92	16.34	16.38	15.87
MFCC_xm	19.41	17.96	16.95	15.96	16.08	15.53
BFCC1BFCC2_xm	18.95	17.76	17.23	17.05	17.24	16.79
BFCC1MFCC_xm	18.91	17.66	17.06	16.62	16.93	16.60
BFCC2MFCC_xm	18.66	17.19	16.45	15.85	15.89	15.25
BFCC1BFCC2MFCC_xm	18.75	17.24	16.80	16.33	16.59	16.44

Table E.4.: Results for all evaluated feature vectors and segment lengths obtained using sampling rate 100.

Segment length	2 s	3 s	4 s	5 s	6 s	7.5 s
BFCC1_xyzm	18.97	17.98	17.46	17.46	17.51	17.43
BFCC2_xyzm	18.02	17.02	16.13	15.98	15.77	15.39
MFCC_xyzm	17.98	16.68	15.27	14.77	15.29	14.83
BFCC1BFCC2_xyzm	17.93	17.14	16.55	16.35	16.39	16.30
BFCC1MFCC_xyzm	17.72	16.63	16.08	15.76	15.99	15.82
BFCC2MFCC_xyzm	17.55	16.17	15.35	14.95	15.02	14.82
BFCC1BFCC2MFCC_xyzm	17.49	16.56	15.91	15.60	15.73	15.65
BFCC1_x	23.68	23.06	22.68	22.51	22.39	22.56
BFCC2_x	24.22	22.76	22.24	22.11	22.29	21.73
MFCC_x	23.68	23.06	22.68	22.51	22.39	22.56
BFCC1BFCC2_x	23.27	22.20	21.48	21.75	21.64	21.42
BFCC1MFCC_x	21.94	20.89	20.79	20.62	20.47	20.26
BFCC2MFCC_x	22.15	20.58	20.09	19.99	19.83	19.28
BFCC1BFCC2MFCC_x	22.04	20.81	20.61	20.54	20.44	20.05
BFCC1_y	25.51	23.50	22.54	21.85	21.03	20.80
BFCC2_y	25.73	23.75	22.51	22.02	21.52	21.15
MFCC_y	22.97	20.54	19.25	19.18	18.59	18.11
BFCC1BFCC2_y	24.62	22.50	21.52	20.83	20.16	19.61
BFCC1MFCC_y	22.99	20.83	19.64	18.81	18.27	17.69
BFCC2MFCC_y	22.81	20.77	19.50	18.89	18.41	18.19
BFCC1BFCC2MFCC_y	22.94	21.00	19.76	18.93	18.10	17.71
BFCC1_z	29.23	28.06	27.24	26.52	26.33	26.29
BFCC2_z	28.40	27.37	26.43	25.74	25.58	25.58
MFCC_z	28.15	26.65	25.96	25.53	25.82	24.99
BFCC1BFCC2_z	28.33	27.25	26.17	25.79	25.30	25.25
BFCC1MFCC_z	28.33	27.25	26.17	25.79	25.30	25.25
BFCC2MFCC_z	27.79	26.49	25.60	25.13	25.05	24.62
BFCC1BFCC2MFCC_z	27.82	26.70	25.73	25.02	24.80	24.50
BFCC1_m	24.45	22.39	21.12	20.27	20.18	19.46
BFCC2_m	23.04	20.66	19.52	18.28	18.33	17.84
MFCC_m	23.12	20.77	19.83	18.55	18.44	18.21
BFCC1BFCC2_m	23.01	20.84	19.40	18.58	18.06	17.90
BFCC1MFCC_m	22.87	20.77	19.66	18.58	18.07	18.07
BFCC2MFCC_m	22.40	20.25	18.78	17.79	17.67	17.43
BFCC1BFCC2MFCC_m	22.45	20.31	19.04	18.03	17.94	17.26
BFCC1_xm	20.45	16.26	18.57	18.71	18.64	18.46
BFCC2_xm	19.44	17.60	16.92	16.42	16.44	15.77
MFCC_xm	19.66	18.12	17.20	16.17	16.31	15.96
BFCC1BFCC2_xm	19.27	17.95	17.31	17.30	17.22	17.07
BFCC1MFCC_xm	19.22	17.94	17.25	16.93	17.07	16.88
BFCC2MFCC_xm	18.77	17.42	16.46	15.90	16.00	15.47
BFCC1BFCC2MFCC_xm	18.77	17.46	16.89	16.56	16.68	16.41

Table E.5.: Results for all evaluated feature vectors and segment lengths obtained using sampling rate 200.

APPENDIX F

SVM RESULTS FOR DATABASE 3

Segment length	2 s		3 s		4 s		5 s		6 s		7.5 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1xyzm BFCC2xyzm MFCCxyzm BFCC1BFCC2xyzm BFCC1MFCCxyzm BFCC2MFCCxyzm BFCC1BFCC2MFCCxyzm	2.69	43.06	2.47	39.31	1.87	40.16	1.31	39.53	0.92	40.81	0.75	40.89
	2.88	43.46	0.38	48.62	2.13	41.33	1.70	41.65	0.86	41.87	1.01	41.71
	2.14	43.15	1.16	43.77	1.29	40.31	1.36	40.86	0.74	43.41	0.73	41.93
	1.47	44.17	1.01	42.25	0.71	41.56	0.42	42.80	1.04	40.57	1.10	40.17
	1.30	44.17	0.80	42.69	0.61	42.35	0.34	43.51	0.20	45.74	0.15	45.82
	1.38	44.63	0.83	43.14	0.69	42.82	0.45	43.86	0.25	46.42	1.19	40.71
	0.76	45.98	0.42	44.85	2.02	38.90	1.66	40.02	0.83	41.08	0.85	40.17
	7.19	52.81	5.46	48.16	4.76	49.71	4.38	48.88	4.02	49.67	3.44	49.80
	5.77	59.36	5.70	53.31	4.24	54.14	4.00	52.92	3.40	51.66	3.38	51.38
	4.30	58.36	3.85	54.91	3.04	53.67	3.01	54.99	2.96	55.94	2.57	51.11
BFCC1BFCC2x BFCC1MFCCx BFCC2MFCCx BFCC1BFCC2MFCCx	4.00	55.73	3.57	50.17	3.04	51.76	2.72	50.79	2.36	50.02	1.99	50.20
	4.13	52.76	2.86	50.47	2.40	50.22	2.17	51.36	1.80	52.72	1.49	52.78
	3.65	59.21	3.10	54.10	2.45	56.01	2.09	55.86	1.74	56.28	1.54	53.55
	2.99	56.63	2.48	52.45	1.97	53.63	1.65	54.28	1.37	55.19	1.05	55.17
	3.96	61.84	3.89	60.02	2.99	56.46	2.81	58.29	2.52	55.94	2.40	54.81
	3.34	65.23	3.63	59.08	3.05	58.95	3.02	59.22	2.82	58.54	2.80	56.89
	3.30	64.06	3.18	60.08	2.53	60.95	2.66	59.82	2.09	61.04	2.21	61.09
	3.01	61.01	2.72	59.05	2.21	57.46	2.02	57.31	2.57	53.92	1.36	57.66
	2.99	61.81	2.43	60.34	2.04	61.48	2.48	56.57	1.77	63.92	1.52	60.91
	2.94	64.59	2.44	61.22	2.12	60.61	1.94	60.47	1.35	61.83	1.60	62.36
BFCC1BFCC2MFCCy BFCC1y BFCC2y MFCCy BFCC1BFCC2y BFCC1MFCCy BFCC2MFCCy BFCC1BFCC2MFCCy	2.55	62.71	1.85	60.63	1.72	61.52	1.27	62.96	1.99	57.31	2.00	54.63
	6.92	68.36	5.92	67.06	5.16	65.06	4.42	63.34	4.36	64.22	4.11	63.53
	5.53	68.82	4.78	65.34	4.33	62.31	3.71	62.66	3.44	65.15	3.85	62.54
	5.07	69.59	5.09	66.83	4.69	67.06	3.88	65.06	3.32	67.31	3.72	61.77
	4.57	67.86	3.67	65.66	5.01	60.10	2.57	64.08	4.40	62.03	2.28	66.24
	4.50	67.82	3.56	66.56	5.25	65.65	4.61	63.72	2.21	67.48	1.79	67.51
	4.47	67.69	3.28	65.43	3.12	64.46	2.45	64.92	1.98	66.38	2.05	65.52
	5.09	66.78	4.52	64.81	4.37	64.78	4.15	62.36	3.36	64.22	1.04	69.86
	5.25	63.05	4.10	63.32	3.59	57.91	3.88	57.09	3.07	55.84	4.71	48.71
	4.67	63.64	4.21	58.59	3.65	54.12	3.45	53.30	2.95	50.33	2.55	55.08
BFCC1BFCC2m BFCC1MFCCm BFCC2MFCCm BFCC1BFCC2MFCCm BFCC1xm BFCC2xm MFCCxm BFCC1BFCC2xm BFCC1MFCCxm BFCC2MFCCxm BFCC1BFCC2MFCCxm	4.93	64.25	4.33	61.71	4.51	58.23	3.48	55.37	2.75	56.86	2.20	56.57
	3.44	60.87	3.89	56.04	2.40	54.91	1.98	56.57	1.76	52.62	1.59	53.77
	3.53	62.26	2.99	59.99	2.33	57.44	2.06	56.52	1.79	53.89	1.62	54.72
	3.46	63.55	3.02	59.35	2.54	56.44	2.29	55.54	1.95	54.64	1.58	56.57
	3.82	59.69	3.56	56.20	1.75	57.37	1.50	58.27	3.26	52.89	2.35	52.37
	3.54	47.66	2.84	44.31	2.18	41.77	1.92	42.88	1.57	41.32	1.40	41.21
	4.88	50.13	2.79	46.20	2.31	46.56	1.60	46.70	1.28	47.14	0.85	47.67
	2.85	48.73	2.01	46.02	1.91	47.20	1.40	46.78	1.08	47.69	0.87	47.58
	3.34	46.73	1.30	48.48	2.64	39.33	2.09	40.75	2.06	40.16	1.71	40.80
	3.34	45.49	1.16	48.33	2.39	40.65	1.96	39.42	1.79	39.95	1.41	40.80
BFCC1BFCC2MFCCxm	3.32	49.24	2.94	43.68	2.52	44.82	2.03	44.05	1.87	44.61	1.36	43.83
	2.67	46.99	2.24	42.17	2.00	41.95	1.45	41.90	1.27	43.72	0.99	43.06

Table F.1.: SVM Results for training with data from setting 0 to 4, sampling rate 25.

Segment length	2 s		3 s		4 s		5 s		6 s		7.5 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1_xyzm	2.66	41.22	0.39	47.03	0.20	49.73	1.26	39.74	1.46	42.50	0.93	41.68
BFCC2_xyzm	2.70	39.44	2.16	40.75	2.00	40.86	1.27	41.36	1.08	41.16	0.85	42.27
MFCC_xyzm	2.31	41.58	1.91	41.33	1.16	41.07	1.36	41.45	0.83	44.03	0.65	41.90
BFCC1BFCC2_xyzm	1.39	41.70	0.80	41.38	0.65	41.13	0.41	42.46	0.29	44.72	1.22	41.18
BFCC1MFCC_xyzm	1.36	41.83	0.68	42.15	0.47	41.58	0.36	43.29	2.00	40.82	1.95	39.73
BFCC2MFCC_xyzm	1.27	41.58	0.60	42.41	0.54	42.92	0.43	44.53	1.26	40.89	1.20	40.63
BFCC1BFCC2MFCC_xyzm	0.61	43.40	0.27	44.71	1.97	38.12	1.26	40.18	1.34	41.91	3.03	38.37
BFCC1_x	5.27	53.23	4.57	48.33	4.45	51.20	3.98	49.56	3.80	49.57	3.57	50.70
BFCC2_x	4.84	53.13	4.65	47.67	4.15	50.99	3.48	50.77	3.52	49.91	3.51	48.34
MFCC_x	4.78	55.76	3.51	53.83	2.84	51.71	2.58	52.86	2.51	51.93	2.45	51.29
BFCC1BFCC2_x	4.01	50.43	3.39	47.36	3.01	49.37	2.56	49.06	2.54	50.12	2.14	50.75
BFCC1MFCC_x	3.72	50.62	2.89	50.02	2.40	49.61	2.05	49.86	1.67	52.51	1.57	53.33
BFCC2MFCC_x	3.94	54.37	3.16	50.76	2.36	53.43	1.78	53.22	1.75	54.36	1.49	55.15
BFCC1BFCC2MFCC_x	3.17	51.70	2.41	49.58	1.94	51.50	1.56	52.09	1.37	54.22	1.12	56.10
BFCC1_y	3.82	62.88	3.45	57.50	2.94	55.77	2.58	56.63	2.60	56.24	2.74	53.11
BFCC2_y	4.58	61.33	2.99	57.68	3.27	59.21	2.97	56.60	2.68	60.14	2.44	57.10
MFCC_y	3.46	64.46	3.50	58.73	2.69	61.31	2.63	60.56	2.24	62.32	2.41	57.78
BFCC1BFCC2_y	3.30	59.33	2.74	56.63	2.42	56.66	2.27	59.30	1.91	59.25	1.85	57.60
BFCC1MFCC_y	2.99	61.30	2.71	59.78	2.13	60.31	1.96	61.28	1.70	61.06	1.61	58.78
BFCC2MFCC_y	3.31	63.35	2.33	60.07	2.18	61.24	2.12	61.14	1.72	64.07	1.64	62.00
BFCC1BFCC2MFCC_y	2.59	62.05	2.09	60.62	1.78	61.39	1.41	62.68	1.13	63.90	1.11	63.40
BFCC1_z	5.88	66.49	5.18	65.43	6.43	64.45	4.27	64.38	4.17	63.90	4.19	64.17
BFCC2_z	5.47	65.70	4.73	62.46	4.30	62.33	4.26	60.34	3.75	60.14	3.91	60.09
MFCC_z	4.58	66.00	4.72	63.53	4.27	64.83	5.65	63.04	3.63	63.52	3.63	64.94
BFCC1BFCC2_z	4.29	66.11	3.30	63.93	3.02	65.90	5.01	57.92	2.00	66.39	1.99	65.94
BFCC1MFCC_z	4.22	64.96	3.57	65.55	5.14	63.69	5.48	63.15	1.96	68.41	1.82	68.71
BFCC2MFCC_z	4.01	65.62	3.23	63.41	2.78	65.83	2.52	65.68	1.81	65.88	2.04	65.85
BFCC1BFCC2MFCC_z	4.51	66.06	2.34	67.22	4.13	65.26	4.03	63.31	3.89	61.95	4.14	60.86
BFCC1_m	4.36	59.05	3.43	57.57	3.95	53.09	3.97	51.90	2.64	49.68	2.44	51.75
BFCC2_m	4.66	56.63	4.42	53.47	3.39	51.33	2.75	52.53	3.88	49.20	2.48	52.24
MFCC_m	4.34	59.70	3.45	55.86	3.35	54.75	2.78	52.64	4.30	49.61	2.76	49.80
BFCC1BFCC2_m	3.49	54.78	3.75	51.90	2.09	51.45	3.40	47.80	1.55	51.86	1.22	54.06
BFCC1MFCC_m	3.39	56.63	3.36	53.74	3.14	51.60	3.30	49.01	1.39	52.07	1.39	52.65
BFCC2MFCC_m	3.56	57.24	2.70	53.64	2.33	53.85	3.53	50.00	1.59	52.58	1.65	52.38
BFCC1BFCC2MFCC_m	3.94	53.47	3.56	51.96	3.07	50.71	3.07	48.62	2.32	48.96	0.88	58.19
BFCC1_xm	2.78	45.52	2.26	40.63	1.81	38.54	1.65	41.42	1.43	40.07	1.12	42.09
BFCC2_xm	2.84	42.56	2.17	41.52	1.87	41.90	1.34	43.78	1.07	42.87	0.81	46.53
MFCC_xm	2.29	45.75	1.65	42.95	1.37	43.13	1.20	43.18	0.81	43.18	0.58	44.67
BFCC1BFCC2_xm	3.17	40.79	2.47	38.44	2.23	39.05	1.75	38.78	1.81	36.92	1.25	40.14
BFCC1MFCC_xm	2.77	40.32	2.06	38.35	1.87	38.63	0.45	48.43	1.51	35.76	1.10	38.68
BFCC2MFCC_xm	2.97	41.48	2.37	39.70	2.06	40.56	1.43	42.19	1.34	41.16	1.11	41.45
BFCC1BFCC2MFCC_xm	2.47	40.81	1.83	39.08	1.45	39.14	1.08	39.77	1.02	39.79	0.79	40.68

Table F.2.: SVM Results for training with data from setting 0 to 4, sampling rate 50.

Segment length	2 s		3 s		4 s		5 s		6 s		7.5 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1_xyzm	0.83	46.11	0.49	49.22	0.28	51.50	1.49	41.38	1.25	42.56	1.21	41.93
BFCC2_xyzm	2.74	39.37	2.22	40.26	1.97	40.62	1.31	40.53	1.13	40.44	0.78	41.93
MFCC_xyzm	2.35	42.16	1.14	42.34	1.22	40.52	1.14	42.95	0.80	44.82	0.73	41.88
BFCC1BFCC2_xyzm	1.47	41.73	0.80	42.29	0.65	42.51	0.47	43.85	0.31	45.81	1.43	41.16
BFCC1MFCC_xyzm	1.37	42.43	0.65	42.92	0.49	42.37	0.39	44.27	0.24	47.52	1.47	41.38
BFCC2MFCC_xyzm	1.35	41.89	0.70	42.35	0.57	42.43	0.47	44.06	0.24	47.52	1.43	41.38
BFCC1BFCC2MFCC_xyzm	0.63	43.74	0.30	45.19	1.58	38.99	1.74	40.75	1.12	42.12	1.04	40.38
BFCC1_x	5.23	54.38	4.79	50.92	4.11	49.76	3.91	49.96	3.83	50.67	3.51	51.89
BFCC2_x	5.16	52.74	4.45	48.64	4.35	50.95	3.55	49.82	3.27	49.20	3.48	49.70
MFCC_x	4.16	56.07	3.58	53.78	3.06	49.33	2.35	53.45	2.87	53.27	2.39	52.48
BFCC1BFCC2_x	3.64	52.60	3.28	49.81	2.92	50.99	2.51	48.94	2.40	50.53	2.08	50.43
BFCC1MFCC_x	3.62	51.51	2.86	51.84	2.27	51.52	1.97	50.87	1.92	52.51	1.41	55.03
BFCC2MFCC_x	3.74	53.84	2.98	51.06	2.29	52.73	1.80	52.74	1.79	53.68	1.75	53.80
BFCC1BFCC2MFCC_x	2.96	52.49	2.26	51.18	1.86	52.58	1.50	52.52	1.39	54.60	1.11	57.16
BFCC1_y	3.71	61.76	3.24	57.77	3.01	56.49	3.01	55.93	2.77	56.03	2.91	52.25
BFCC2_y	4.18	62.89	3.61	59.33	3.44	59.93	2.92	58.12	2.64	60.58	2.90	59.03
MFCC_y	3.36	63.43	2.91	60.70	2.33	61.07	2.28	61.37	2.12	62.15	2.32	59.25
BFCC1BFCC2_y	3.09	59.61	2.64	56.45	2.42	57.36	2.25	58.29	1.90	59.04	1.98	57.12
BFCC1MFCC_y	2.98	61.46	2.96	59.53	2.44	62.14	2.05	59.47	1.78	61.23	1.57	59.25
BFCC2MFCC_y	3.11	62.94	2.31	60.39	2.21	61.10	2.07	61.09	1.66	65.06	1.65	61.39
BFCC1BFCC2MFCC_y	2.37	61.88	2.10	60.85	1.81	61.88	1.47	61.67	2.34	58.05	1.23	63.85
BFCC1_z	5.77	66.21	4.96	64.93	4.76	65.68	7.36	60.54	4.02	64.55	4.31	63.30
BFCC2_z	5.43	65.90	5.07	62.01	4.74	60.63	4.17	58.18	4.15	60.38	3.85	58.80
MFCC_z	5.37	65.42	5.13	63.41	4.24	65.05	3.99	63.90	3.91	63.76	3.40	64.21
BFCC1BFCC2_z	4.53	64.55	3.58	64.02	3.10	64.88	2.39	63.93	4.99	58.05	2.02	65.30
BFCC1MFCC_z	3.98	66.90	3.64	65.96	2.60	67.70	2.19	67.86	2.01	68.24	1.92	68.62
BFCC2MFCC_z	4.18	65.99	3.45	63.82	2.89	65.11	2.56	65.44	1.85	66.05	1.56	66.71
BFCC1BFCC2MFCC_z	3.40	66.39	2.47	66.96	2.00	68.85	3.97	62.52	3.76	62.77	3.92	61.85
BFCC1_m	5.07	59.81	4.69	59.59	4.52	54.96	5.14	53.67	4.28	51.01	2.65	54.43
BFCC2_m	4.05	58.84	4.72	53.47	3.78	50.50	2.71	51.53	2.63	50.26	2.69	52.30
MFCC_m	4.14	61.91	3.69	56.75	3.24	56.23	2.99	53.81	4.15	50.77	2.74	49.84
BFCC1BFCC2_m	4.37	56.01	2.81	53.41	2.40	53.05	3.51	50.70	3.42	49.23	3.26	50.70
BFCC1MFCC_m	4.48	57.55	2.52	56.05	2.20	56.98	3.60	50.37	1.40	55.21	1.46	55.16
BFCC2MFCC_m	3.20	59.25	2.77	52.89	2.40	53.43	3.79	50.29	1.47	53.57	1.57	53.25
BFCC1BFCC2MFCC_m	3.55	56.36	2.08	55.23	3.20	53.13	2.67	51.94	2.46	51.49	0.91	59.16
BFCC1_xm	3.26	46.08	2.52	45.82	2.06	42.73	1.92	43.94	1.50	45.74	1.14	47.29
BFCC2_xm	3.69	41.35	2.23	40.97	1.76	41.39	1.30	43.44	2.72	38.39	0.84	47.43
MFCC_xm	3.18	44.96	1.74	42.54	1.38	43.47	1.00	44.02	0.85	44.31	0.73	43.57
BFCC1BFCC2_xm	2.95	41.59	1.18	45.05	2.19	38.59	1.83	39.04	1.88	37.40	1.34	41.88
BFCC1MFCC_xm	2.91	41.46	2.11	40.27	1.87	39.20	1.75	38.66	1.64	37.74	1.23	40.47
BFCC2MFCC_xm	2.80	42.04	0.93	46.10	1.94	39.41	1.41	41.74	1.40	41.37	1.21	41.88
BFCC1BFCC2MFCC_xm	2.42	41.20	1.88	39.27	1.44	39.52	1.13	41.30	1.12	41.30	0.84	42.20

Table F.3.: SVM Results for training with data from setting 0 to 4, sampling rate 100.

Segment length	2 s		3 s		4 s		5 s		6 s		7.5 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1_xyzm	2.99	41.66	2.83	41.31	0.38	51.16	1.46	41.75	1.71	42.09	1.28	40.86
	2.92	38.69	2.23	40.37	1.91	41.18	1.31	40.63	1.12	40.89	0.86	41.59
MFCC_xyzm	2.51	42.45	1.18	42.48	1.20	40.94	1.30	42.52	0.86	45.03	0.72	42.82
	1.61	41.38	0.87	41.82	0.73	42.20	2.68	39.83	1.55	40.51	1.46	40.91
BFCC1BFCC2_xyzm	1.50	42.00	0.80	42.88	0.54	42.73	0.41	44.12	1.45	42.02	1.42	41.23
	1.51	41.87	0.66	42.02	0.54	43.02	1.59	40.38	1.30	40.68	1.36	40.41
BFCC2MFCC_xyzm	0.72	43.68	0.31	44.79	0.20	46.70	1.76	40.19	1.12	40.96	1.21	39.09
	5.77	53.89	4.96	51.38	4.20	49.90	4.09	49.34	3.99	50.02	3.75	52.45
BFCC1BFCC2MFCC_xyzm	4.70	53.71	4.32	48.67	4.14	50.29	3.74	49.37	3.16	48.79	3.43	47.64
	4.60	54.44	3.43	52.90	3.05	48.91	2.47	52.61	3.03	52.48	2.25	52.23
MFCC_x	3.62	52.19	3.18	50.69	2.77	50.24	2.34	49.78	2.30	49.98	2.20	49.18
	3.75	52.20	2.82	53.12	2.35	51.20	1.95	51.62	1.94	53.47	1.55	54.36
BFCC1MFCC_x	3.57	54.09	2.96	51.79	2.34	51.96	1.91	52.20	1.99	52.34	1.51	54.14
	2.86	53.30	2.24	52.07	1.79	53.17	1.46	52.36	1.43	52.65	1.12	55.77
BFCC1BFCC2MFCC_x	4.08	60.70	3.31	59.78	3.24	57.32	3.28	58.25	2.93	55.93	3.03	56.00
	4.12	62.33	3.57	59.74	3.52	60.29	3.03	57.81	2.58	60.17	2.28	59.18
MFCC_y	3.55	63.40	2.86	60.55	2.89	59.48	2.68	61.57	2.10	61.78	2.12	58.55
	3.35	59.22	2.74	57.89	2.32	58.27	2.32	57.89	1.97	59.15	1.94	58.86
BFCC1BFCC2_y	3.36	60.65	2.38	60.18	2.54	61.52	2.15	59.87	1.94	61.57	1.75	60.45
	3.20	62.65	2.51	61.43	2.19	61.56	2.04	61.05	1.69	64.03	1.52	62.64
BFCC1MFCC_y	3.42	59.67	2.15	61.59	1.79	62.71	1.57	61.96	2.29	58.32	2.50	57.18
	5.92	66.12	4.93	64.79	4.13	65.60	5.96	62.04	3.68	64.62	3.63	64.23
BFCC1_z	5.39	64.29	4.79	62.96	4.68	60.95	4.29	58.49	3.92	60.72	4.09	59.00
	5.19	66.69	5.18	62.73	4.10	64.92	3.89	63.96	3.49	62.74	3.31	64.95
MFCC_z	4.38	64.64	3.59	63.62	2.80	63.73	2.48	63.36	4.58	59.28	1.86	64.23
	4.50	65.73	3.50	65.36	2.33	67.51	2.14	67.51	1.91	67.56	1.82	68.41
BFCC1BFCC2_z	4.31	65.65	3.43	64.79	2.78	64.71	2.66	64.71	1.83	66.12	1.95	66.00
	3.43	65.66	2.45	66.84	1.83	67.47	3.78	63.14	3.57	62.63	3.51	61.95
BFCC2MFCC_z	5.70	60.52	5.02	62.88	5.15	58.25	5.13	57.12	3.66	56.17	3.13	56.23
	4.76	58.04	4.08	52.47	3.67	50.67	2.81	51.84	2.70	49.85	2.83	52.32
BFCC1_m	4.12	61.45	3.55	56.66	3.27	56.98	3.09	54.65	4.14	50.91	2.77	50.36
	3.65	57.07	2.98	54.04	3.39	51.03	3.70	51.13	3.59	49.27	1.60	55.91
BFCC2_m	3.46	59.46	2.50	57.23	3.21	55.45	3.81	52.72	1.43	56.48	1.38	55.64
	3.33	59.37	2.81	53.13	2.43	54.75	2.26	53.99	4.27	49.98	1.61	53.41
BFCC2MFCC_m	3.65	56.08	2.16	56.23	3.24	53.22	2.77	52.78	2.56	52.00	0.95	59.86
	3.33	46.82	2.73	44.97	2.16	43.45	2.05	45.19	1.62	46.70	1.33	47.50
BFCC1_xm	2.75	42.57	2.17	40.95	1.80	41.69	1.33	44.01	2.86	38.43	0.82	46.68
	3.23	44.62	1.75	42.73	1.33	43.00	1.16	43.51	0.94	43.79	0.68	43.68
BFCC2_xm	3.10	41.35	2.46	39.82	2.22	38.14	1.97	39.00	1.85	37.74	1.65	40.68
	2.92	42.16	2.19	40.03	1.88	38.73	1.86	38.40	1.71	37.95	1.53	38.68
BFCC1BFCC2_xm	2.85	42.36	2.41	38.72	2.00	39.50	1.45	42.28	1.43	41.40	1.26	41.36
	2.48	41.72	1.86	40.17	1.46	39.86	1.28	40.24	1.13	41.54	0.87	41.77
BFCC2MFCC_xm												

Table F.4.: SVM Results for training with data from setting 0 to 4, sampling rate 200.

APPENDIX G

K-NN RESULTS FOR DATABASE 3

Segment length	2 s		3 s		4 s		5 s		6 s		7.5 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1_xyzm	5.33	32.93	5.36	28.79	5.74	26.15	5.88	25.10	6.12	24.65	6.44	24.18
BFCC2_xyzm	4.95	33.13	5.01	27.87	4.99	24.15	5.21	23.00	5.50	21.43	5.60	21.24
MFCC_xyzm	4.63	32.69	4.89	26.73	4.90	23.75	4.99	22.42	5.17	21.84	5.53	21.19
BFCC1BFCC2_xyzm	5.06	30.51	5.15	26.15	5.25	24.07	5.51	23.84	5.85	22.60	6.04	21.42
BFCC1MFCC_xyzm	5.00	29.94	5.10	25.19	5.30	23.22	5.55	23.27	5.81	21.43	5.98	20.74
BFCC2MFCC_xyzm	4.66	30.48	4.94	24.69	4.90	22.22	5.05	21.85	5.44	20.27	5.61	19.34
BFCC1BFCC2MFCC_xyzm	4.88	29.51	5.03	24.69	5.17	22.28	5.47	22.56	5.77	20.58	5.84	19.88

Table G.1.: k-NN Results for training with data from setting 0 to 9, sampling rate 25.

Segment length	2 s		3 s		4 s		5 s		6 s		7.5 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1_xyzm	5.49	30.21	5.68	26.79	6.25	23.91	6.28	23.98	6.67	23.15	7.14	23.31
BFCC2_xyzm	5.22	28.75	5.40	25.16	5.53	21.92	5.48	22.47	5.74	20.17	6.19	18.55
MFCC_xyzm	5.22	29.04	5.40	24.03	5.90	21.72	5.73	20.05	5.76	19.86	6.28	16.46
BFCC1BFCC2_xyzm	5.22	27.39	5.57	24.40	5.77	22.49	5.89	22.72	6.47	21.98	6.69	21.04
BFCC1MFCC_xyzm	5.35	26.87	5.53	23.68	6.12	21.13	6.10	20.35	6.27	20.03	6.68	18.37
BFCC2MFCC_xyzm	5.12	26.58	5.54	23.14	5.66	20.54	5.60	19.64	5.87	18.53	6.28	17.10
BFCC1BFCC2MFCC_xyzm	5.22	25.69	5.47	22.88	5.80	20.90	5.86	20.35	6.30	19.79	6.49	18.46

Table G.2.: k-NN Results for training with data from setting 0 to 9, sampling rate 50.

Segment length	2 s		3 s		4 s		5 s		6 s		7.5 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1_xyzm	5.57	31.86	5.62	29.14	5.97	25.91	6.12	25.65	6.43	25.98	6.90	24.51
BFCC2_xyzm	5.22	28.88	5.46	25.26	5.58	22.43	5.40	22.24	5.71	20.17	6.18	19.42
MFCC_xyzm	5.42	29.16	5.49	24.35	5.77	22.23	5.42	20.76	5.63	19.73	6.11	16.83
BFCC1BFCC2_xyzm	5.25	28.26	5.37	25.74	5.62	23.30	5.56	23.40	6.10	23.08	6.53	21.33
BFCC1MFCC_xyzm	5.42	27.52	5.42	24.44	5.77	22.17	5.68	21.14	6.11	20.58	6.63	18.10
BFCC2MFCC_xyzm	5.18	26.57	5.51	23.02	5.55	20.39	5.47	19.69	5.78	17.44	6.26	17.10
BFCC1BFCC2MFCC_xyzm	5.22	26.38	5.32	23.56	5.56	21.38	5.47	20.68	5.97	20.58	6.39	19.01

Table G.3.: k-NN Results for training with data from setting 0 to 9, sampling rate 100.

Segment length	2 s		3 s		4 s		5 s		6 s		7.5 s	
	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR	FMR	FNMR
BFCC1_xyzm	5.72	32.18	5.78	29.33	6.14	26.88	6.14	25.81	6.73	25.64	7.26	23.64
BFCC2_xyzm	5.17	29.14	5.52	24.88	5.82	22.53	5.42	21.85	5.68	20.75	6.12	19.59
MFCC_xyzm	5.50	28.86	5.50	24.97	5.77	22.36	5.46	20.78	5.75	19.66	6.14	17.32
BFCC1BFCC2_xyzm	5.31	28.58	5.55	26.05	5.92	23.51	5.72	23.25	6.34	22.97	6.65	21.68
BFCC1MFCC_xyzm	5.45	27.43	5.37	24.94	5.89	22.19	5.63	20.86	6.17	20.72	6.83	18.14
BFCC2MFCC_xyzm	5.24	26.50	5.64	23.09	5.85	20.20	5.47	19.63	5.77	18.36	6.16	17.64
BFCC1BFCC2MFCC_xyzm	5.28	26.44	5.42	23.97	5.81	20.94	5.53	20.81	6.02	20.14	6.45	19.23

Table G.4.: k-NN Results for training with data from setting 0 to 9, sampling rate 200.

APPENDIX H

FEATURE SETS BASED ON DISCRIMINATIVE POTENTIAL SCORE

Table H.1.: Feature sets depending on the DPS. A 1 indicates that the feature is included in the corresponding feature vector, a 0 indicates that the feature is not included.

Feature	minimal score necessary for the feature set											
	> -0.4	> -0.3	> -0.2	> -0.1	> 0	> 0.1	> 0.2	> 0.3	> 0.4	> 0.5	> 0.6	> 0.7
bfccl.x(1)	1	1	1	1	1	1	1	1	1	1	0	0
bfccl.x(2)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.x(3)	1	1	1	1	1	1	1	1	1	1	0	0
bfccl.x(4)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.x(5)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.x(6)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.x(7)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.x(8)	1	1	1	1	1	1	1	0	0	0	0	0
bfccl.x(9)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.x(10)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.x(11)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.x(12)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.x(13)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.y(1)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.y(2)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.y(3)	1	1	1	1	1	1	1	1	1	1	0	0
bfccl.y(4)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.y(5)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.y(6)	1	1	1	1	1	1	1	0	0	0	0	0
bfccl.y(7)	1	1	1	1	1	0	0	0	0	0	0	0
bfccl.y(8)	1	1	1	1	1	1	1	0	0	0	0	0
bfccl.y(9)	1	1	1	1	1	1	1	0	0	0	0	0
bfccl.y(10)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.y(11)	1	1	1	1	1	1	0	0	0	0	0	0
bfccl.y(12)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.y(13)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.z(1)	1	1	1	1	1	1	1	1	1	1	1	1
bfccl.z(2)	1	1	1	1	1	1	1	1	1	1	0	0
bfccl.z(3)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.z(4)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.z(5)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.z(6)	1	1	1	1	1	1	1	1	1	1	0	0
bfccl.z(7)	1	1	1	1	1	1	1	1	0	0	0	0
bfccl.z(8)	1	1	1	1	1	1	1	1	1	0	0	0
bfccl.z(9)	1	1	1	1	1	1	1	0	0	0	0	0
bfccl.z(10)	1	1	1	1	1	1	0	0	0	0	0	0
bfccl.z(11)	1	1	1	1	1	1	1	0	0	0	0	0
bfccl.z(12)	1	1	1	1	1	1	1	1	1	1	0	0
bfccl.z(13)	1	1	1	1	1	1	1	1	0	0	0	0

continued on next page

H. Feature Sets Based on Discriminative Potential Score

<i>continued from previous page</i>												
Feature	minimal score necessary for the feature set											
	> -0.4	> -0.3	> -0.2	> -0.1	> 0	> 0.1	> 0.2	> 0.3	> 0.4	> 0.5	> 0.6	> 0.7
bfee1_m(1)	1	1	1	1	1	1	1	1	1	1	1	0
bfee1_m(2)	1	1	1	1	1	1	1	0	0	0	0	0
bfee1_m(3)	1	1	1	1	1	1	1	0	0	0	0	0
bfee1_m(4)	1	1	1	1	1	1	1	1	1	1	1	0
bfee1_m(5)	1	1	1	1	1	1	1	1	0	0	0	0
bfee1_m(6)	1	1	1	1	1	1	1	0	0	0	0	0
bfee1_m(7)	1	1	1	1	1	1	0	0	0	0	0	0
bfee1_m(8)	1	1	1	1	1	1	1	1	1	0	0	0
bfee1_m(9)	1	1	1	1	1	1	0	0	0	0	0	0
bfee1_m(10)	1	1	1	1	1	1	1	0	0	0	0	0
bfee1_m(11)	1	1	1	1	1	1	1	1	0	0	0	0
bfee1_m(12)	1	1	1	1	1	1	1	1	0	0	0	0
bfee1_m(13)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_x(1)	1	1	1	1	1	1	1	1	1	1	1	1
bfee2_x(2)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_x(3)	1	1	1	1	1	1	1	1	1	1	0	0
bfee2_x(4)	1	1	1	1	1	1	1	1	1	0	0	0
bfee2_x(5)	1	1	1	1	1	1	1	1	1	0	0	0
bfee2_x(6)	1	1	1	1	1	1	1	1	1	0	0	0
bfee2_x(7)	1	1	1	1	1	1	1	1	1	1	0	0
bfee2_x(8)	1	1	1	1	1	1	1	1	1	0	0	0
bfee2_x(9)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_x(10)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_x(11)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_x(12)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_x(13)	1	1	1	1	1	1	0	0	0	0	0	0
bfee2_y(1)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_y(2)	1	1	1	1	1	1	1	1	1	1	0	0
bfee2_y(3)	1	1	1	1	1	1	1	1	1	1	0	0
bfee2_y(4)	1	1	1	1	1	1	1	1	1	0	0	0
bfee2_y(5)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_y(6)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_y(7)	1	1	1	1	1	0	0	0	0	0	0	0
bfee2_y(8)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_y(9)	1	1	1	1	1	1	1	1	1	0	0	0
bfee2_y(10)	1	1	1	1	1	0	0	0	0	0	0	0
bfee2_y(11)	1	1	1	1	1	0	0	0	0	0	0	0
bfee2_y(12)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_y(13)	1	1	1	1	1	1	1	1	1	0	0	0
bfee2_z(1)	1	1	1	1	1	1	1	1	1	1	1	1
bfee2_z(2)	1	1	1	1	1	1	1	1	1	1	1	0
bfee2_z(3)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_z(4)	1	1	1	1	1	1	1	1	1	1	0	0
bfee2_z(5)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_z(6)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_z(7)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_z(8)	1	1	1	1	1	1	1	1	1	1	0	0
bfee2_z(9)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_z(10)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_z(11)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_z(12)	1	1	1	1	0	0	0	0	0	0	0	0
bfee2_z(13)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_m(1)	1	1	1	1	1	1	1	1	1	1	1	0
bfee2_m(2)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_m(3)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_m(4)	1	1	1	1	1	1	1	1	1	1	0	0
bfee2_m(5)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_m(6)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_m(7)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_m(8)	1	1	1	1	1	1	1	1	0	0	0	0
bfee2_m(9)	1	1	1	1	1	1	1	0	0	0	0	0
bfee2_m(10)	1	1	1	1	1	1	1	1	1	1	0	0
bfee2_m(11)	1	1	1	1	1	0	0	0	0	0	0	0
bfee2_m(12)	1	1	1	1	1	0	0	0	0	0	0	0
bfee2_m(13)	1	1	1	1	1	0	0	0	0	0	0	0
mfcc_x(1)	1	1	1	1	1	1	1	1	1	1	1	1
mfcc_x(2)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc_x(3)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc_x(4)	1	1	1	1	1	1	1	1	1	1	1	0
mfcc_x(5)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc_x(6)	1	1	1	1	1	1	1	1	0	0	0	0
mfcc_x(7)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc_x(8)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc_x(9)	1	1	1	1	1	1	1	0	0	0	0	0
mfcc_x(10)	1	1	1	1	1	0	0	0	0	0	0	0
mfcc_x(11)	1	1	1	1	1	1	1	0	0	0	0	0
mfcc_x(12)	1	1	1	1	1	1	1	0	0	0	0	0
mfcc_x(13)	1	1	1	1	1	1	0	0	0	0	0	0

continued on next page

<i>continued from previous page</i>												
Feature	minimal score necessary for the feature set											
	> -0.4	> -0.3	> -0.2	> -0.1	> 0	> 0.1	> 0.2	> 0.3	> 0.4	> 0.5	> 0.6	> 0.7
mfcc.y(1)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc.y(2)	1	1	1	1	1	1	1	1	1	1	1	0
mfcc.y(3)	1	1	1	1	1	1	1	1	1	1	0	0
mfcc.y(4)	1	1	1	1	1	1	1	1	0	0	0	0
mfcc.y(5)	1	1	1	1	1	1	1	1	0	0	0	0
mfcc.y(6)	1	1	1	1	1	1	1	0	0	0	0	0
mfcc.y(7)	1	1	1	1	1	1	1	1	1	1	0	0
mfcc.y(8)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc.y(9)	1	1	1	1	1	1	1	1	0	0	0	0
mfcc.y(10)	1	1	1	1	1	0	0	0	0	0	0	0
mfcc.y(11)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.y(12)	1	1	1	1	1	0	0	0	0	0	0	0
mfcc.y(13)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.z(1)	1	1	1	1	1	1	1	1	1	1	1	0
mfcc.z(2)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc.z(3)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.z(4)	1	1	1	1	1	1	1	1	1	1	0	0
mfcc.z(5)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc.z(6)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc.z(7)	1	1	1	1	1	1	1	1	1	0	0	0
mfcc.z(8)	1	1	1	1	1	1	1	1	1	1	0	0
mfcc.z(9)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.z(10)	1	1	1	1	1	1	1	1	0	0	0	0
mfcc.z(11)	1	1	1	1	1	1	1	0	0	0	0	0
mfcc.z(12)	1	1	1	1	1	1	1	0	0	0	0	0
mfcc.z(13)	1	1	1	1	1	0	0	0	0	0	0	0
mfcc.m(1)	1	1	1	1	1	1	1	1	1	1	1	0
mfcc.m(2)	1	1	1	1	1	1	1	0	0	0	0	0
mfcc.m(3)	1	1	1	1	1	1	1	1	1	1	1	0
mfcc.m(4)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.m(5)	1	1	1	1	1	1	1	1	0	0	0	0
mfcc.m(6)	1	1	1	1	1	1	1	1	0	0	0	0
mfcc.m(7)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.m(8)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.m(9)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.m(10)	1	1	1	1	1	1	1	0	0	0	0	0
mfcc.m(11)	1	1	1	1	1	0	0	0	0	0	0	0
mfcc.m(12)	1	1	1	1	1	1	0	0	0	0	0	0
mfcc.m(13)	1	1	1	1	1	0	0	0	0	0	0	0
mean.x	1	0	0	0	0	0	0	0	0	0	0	0
mean.y	1	1	1	1	1	1	0	0	0	0	0	0
mean.z	1	1	1	1	1	0	0	0	0	0	0	0
mean.m	1	1	1	1	1	1	1	1	1	1	1	1
std.x	1	1	1	1	1	1	1	1	1	1	1	1
std.y	1	1	1	1	1	1	1	1	1	1	1	0
std.z	1	1	1	1	1	1	1	1	1	1	1	1
std.m	1	1	1	1	1	1	1	1	1	1	1	1
max.x	1	1	1	1	1	1	1	1	1	1	1	1
max.y	1	1	1	1	1	1	1	0	0	0	0	0
max.z	1	1	1	1	1	1	1	1	1	1	0	0
max.m	1	1	1	1	1	1	1	1	1	1	1	0
min.x	1	1	1	1	1	1	1	0	0	0	0	0
min.y	1	1	1	1	1	1	1	0	0	0	0	0
min.z	1	1	1	1	1	1	1	1	1	1	0	0
min.m	1	1	1	1	1	0	0	0	0	0	0	0
rms.x	1	1	1	1	1	1	1	1	1	1	1	1
rms.y	1	1	1	1	1	1	1	1	1	1	1	0
rms.z	1	1	1	1	1	1	1	1	1	1	1	1
rms.m	1	1	1	1	1	1	1	1	1	1	1	1
cros.x	1	1	1	1	1	1	1	0	0	0	0	0
cros.y	1	1	1	1	1	1	1	1	1	0	0	0
cros.z	1	1	1	1	1	1	1	1	1	0	0	0
cros.m	0	0	0	0	0	0	0	0	0	0	0	0
diff.x	1	1	0	0	0	0	0	0	0	0	0	0
diff.y	1	1	0	0	0	0	0	0	0	0	0	0
diff.z	1	1	1	1	0	0	0	0	0	0	0	0
diff.m	1	1	1	0	0	0	0	0	0	0	0	0
bin-10.x(1)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10.x(2)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10.x(3)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10.x(4)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10.x(5)	1	1	1	1	1	1	1	0	0	0	0	0
bin-10.x(6)	1	1	1	1	1	1	1	0	0	0	0	0
bin-10.x(7)	1	1	1	1	1	1	0	0	0	0	0	0
bin-10.x(8)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10.x(9)	1	1	1	0	0	0	0	0	0	0	0	0
bin-10.x(10)	1	1	1	1	1	1	1	1	0	0	0	0
bin-10.y(1)	1	1	1	0	0	0	0	0	0	0	0	0

continued on next page

H. Feature Sets Based on Discriminative Potential Score

<i>continued from previous page</i>												
Feature	minimal score necessary for the feature set											
	> -0.4	> -0.3	> -0.2	> -0.1	> 0	> 0.1	> 0.2	> 0.3	> 0.4	> 0.5	> 0.6	> 0.7
bin-10-y(2)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-y(3)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10-y(4)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-y(5)	1	1	1	0	0	0	0	0	0	0	0	0
bin-10-y(6)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-y(7)	1	1	1	0	0	0	0	0	0	0	0	0
bin-10-y(8)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10-y(9)	1	1	1	1	1	1	0	0	0	0	0	0
bin-10-y(10)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-z(1)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10-z(2)	1	1	1	1	1	1	1	1	0	0	0	0
bin-10-z(3)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10-z(4)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-z(5)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-z(6)	1	1	1	0	0	0	0	0	0	0	0	0
bin-10-z(7)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-z(8)	1	1	1	1	1	1	1	1	1	0	0	0
bin-10-z(9)	1	1	1	1	1	1	1	0	0	0	0	0
bin-10-z(10)	1	1	1	1	1	1	1	1	0	0	0	0
bin-10-m(1)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10-m(2)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-m(3)	1	1	1	1	1	1	1	0	0	0	0	0
bin-10-m(4)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10-m(5)	1	1	1	1	1	0	0	0	0	0	0	0
bin-10-m(6)	1	1	1	1	1	1	1	0	0	0	0	0
bin-10-m(7)	1	1	1	1	1	1	0	0	0	0	0	0
bin-10-m(8)	1	1	1	1	1	1	0	0	0	0	0	0
bin-10-m(9)	1	1	1	1	0	0	0	0	0	0	0	0
bin-10-m(10)	0	0	0	0	0	0	0	0	0	0	0	0
bin-5-x(1)	1	1	1	1	1	1	1	0	0	0	0	0
bin-5-x(2)	1	1	1	1	1	1	0	0	0	0	0	0
bin-5-x(3)	1	1	1	1	1	1	1	1	1	0	0	0
bin-5-x(4)	1	1	1	1	1	1	0	0	0	0	0	0
bin-5-x(5)	1	1	1	1	1	1	1	1	1	0	0	0
bin-5-y(1)	1	1	1	1	1	0	0	0	0	0	0	0
bin-5-y(2)	1	1	1	1	1	0	0	0	0	0	0	0
bin-5-y(3)	1	1	1	1	1	0	0	0	0	0	0	0
bin-5-y(4)	1	1	1	1	1	0	0	0	0	0	0	0
bin-5-y(5)	1	1	1	1	0	0	0	0	0	0	0	0
bin-5-z(1)	1	1	1	1	1	1	1	0	0	0	0	0
bin-5-z(2)	1	1	1	0	0	0	0	0	0	0	0	0
bin-5-z(3)	1	1	1	0	0	0	0	0	0	0	0	0
bin-5-z(4)	1	1	1	1	1	1	1	1	1	0	0	0
bin-5-z(5)	1	1	1	1	1	1	1	1	0	0	0	0
bin-5-m(1)	1	1	1	1	1	0	0	0	0	0	0	0
bin-5-m(2)	1	1	1	1	1	1	0	0	0	0	0	0
bin-5-m(3)	1	1	1	1	1	1	1	1	0	0	0	0
bin-5-m(4)	1	1	1	1	1	1	0	0	0	0	0	0
bin-5-m(5)	0	0	0	0	0	0	0	0	0	0	0	0

ANOVA Analysis of Variance 24

API Application Programming Interface 37

BFCC Bark Frequency Cepstral Coefficient; These features are described in section 5.3.1.
40, 41, 49, 55, 58, 61–64, 67–71, 73–85, 87, 90–98, 100, 108, 111, 121

CASED Center for Advanced Security Research Darmstadt 26, 50, 113

CCR Correct Classification Rate 18, 21

CMS Cumulative Match Score 18, 19

CRM Cyclic Rotation Metric 43, 103–108

DAS Draw-A-Secret 6

DCT Discrete Cosine Transform 41

DET Detection Error Trade Off; A DET-curve is the plot of FMR versus FNMR for various thresholds.
48, 61, 78, 81–83, 91, 92, 105, 106

DFT Discrete Fourier Transform 40

DPS Discriminative Potential Score 41, 42, 59, 73–75, 85

DTW Dynamic Time Warping 18, 23, 24, 39, 42, 43, 103

EER Equal Error Rate; The rate where the FMR equals the FNMR.
8, 9, 18–24, 48, 57, 61–65, 69, 78, 80–85, 90–93, 98, 105, 107, 108, 110–113, 121, 122

- FAR** False Accept Rate; Proportion of verification transactions with wrongful claims of identity that are incorrectly confirmed.
19, 22
- FFSM** Fuzzy Finite State Machine 24
- FFT** Fast Fourier Transform 19, 20, 24
- FMR** False Match Rate; Proportion of the completed biometric non-match comparison trials that result in a false match, where a false match is a comparison decision of “match” for a biometric probe and a biometric reference that are from different biometric capture subjects.
25, 47, 48, 52–57, 65, 67–69, 71, 74–76, 92–94, 96–103, 106, 107, 110, 144, 145
- FNMR** False Non Match Rate; Proportion of the completed biometric match comparison trials that result in a false non-match, where a false non-match is a comparison decision of “non-match” for a biometric probe and a biometric reference that are from the same biometric capture subject and of the same biometric characteristic.
25, 47, 48, 52–57, 65, 67–69, 71, 73–76, 85, 86, 92–94, 96–99, 101–103, 106, 107, 110, 112, 144, 145
- FRR** False Reject Rate; Proportion of verification transactions with truthful claims of identity that are incorrectly denied.
19, 22
- GMR** Genuine Match Rate; Proportion of verification transactions with truthful claims of identity that are correctly confirmed (1-FNMR).
20, 21, 25
- GPS** Global Positioning System 24
- GUI** Graphical User Interface 31, 32
- HMM** Hidden Markov Model; The approach is described in section 5.4.2.1.
4, 8, 18, 19, 30, 43–45, 50, 52–54, 57–59, 61, 62, 65, 67, 69, 77, 82, 85, 87, 88, 90, 92, 94, 97, 98, 103, 105, 107, 108, 110–113
- HTER** Half Total Error Rate; The rate is computed as half of the sum of FMR and FNMR.
8, 48, 52, 53, 55, 57, 65, 67–69, 71, 73–76, 85, 95–98, 102, 107, 108, 112, 113
- HTK** Hidden Markov Model Toolkit 44, 51
- IBL** Instance Based Learning 47
- k-NN** k-Nearest Neighbour; The algorithm is described in section 5.4.2.3.
4, 43, 47, 59, 73, 76, 77, 85–87, 103, 108–113
- LED** Light Emitting Diode 9
- LFCC** Linear Frequency Cepstral Coefficient; These features are described in section 5.3.1.
40, 42

- MBASSy** Modular Biometric Authentication Service System 26–33, 77, 86, 103, 112
- MFCC** Mel Frequency Cepstral Coefficient; These features are described in section 5.3.1.
8, 40, 41, 49, 55, 58, 61–64, 67–69, 73–75, 77, 78, 80–85, 87, 90–98, 100–102, 104, 108, 111, 121
- MV** Majority Voting 105, 107, 108
- NFC** Near Field Communication 26
- OS** Operating System 3, 59
- PCA** Principal Component Analysis 23, 41, 84
- PIN** Personal Identification Number 1, 3, 5, 11–14, 26, 107
- RDTW** Rotation Dynamic Time Warping 43, 111
- RR** Recognition Rate 19
- SIM** Subscriber Identity Module 5
- SVM** Support Vector Machine; The approach is described in section 5.4.2.2.
4, 24, 30, 43, 45, 46, 50, 54, 57–59, 65, 69–73, 76, 85, 87, 94, 96, 98, 105, 107, 108, 110–113
- WEKA** Waikato Environment for Knowledge Analysis 47, 73
- WISDM** Wireless Sensor Data Mining 24

- [1] MOBIO Project. <http://www.mobioproject.org/>. [online; last accessed 2012-03-14].
- [2] The Science Behind Passfaces. Technical report, Passfaces Corporation, <http://www.passfaces.com/published/The%20Science%20Behind%20Passfaces.pdf>. [online; last accessed 2012-03-14].
- [3] M. D. Addlesee, A. Jones, F. Livesey, and F. Samaria. The ORL active floor. *IEEE Personal Communications*, 4(5):35–41, 1997.
- [4] J. K. Aggarwal and Q. Cai. Human Motion Analysis: A Review. *Computer Vision and Image Understanding*, 73:428–440, 1999.
- [5] D. W. Aha, D. Kibler, and M. K. Albert. Instance-Based Learning Algorithms. In *Machine Learning*, volume 6, pages 37–66, 1991.
- [6] H. J. Ailisto, M. Lindholm, J. Mäntyjärvi, E. Vildjiounaite, and S. Mäkelä. Identifying people from gait pattern with accelerometers. *Biometric Technology for Human Identification II*, 5779(1):7–14, 2005. VTT Electronics, Finland.
- [7] N. Akae, Y. Makihara, and Y. Yagi. The optimal camera arrangement by a performance model for gait recognition. In *IEEE International Conference on Automatic Face and Gesture Recognition and Workshops*, pages 292 –297, March 2011.
- [8] R. Allen, editor. *English Dictionary*. Penguin Group, 2004.
- [9] G. Ariyanto and M. Nixon. Model-Based 3D Gait Biometrics. In *International Joint Conference on Biometrics*, October 2011.
- [10] Asahi Kasei EMD Corporation. Press Release: World’s smallest and thinnest 6-axis electronic compass - combination of 3-axis geomagnetic sensor with 3-axis acceleration sensor - . <http://www.asahi-kasei.co.jp/asahi/en/news/2005/e060322.html>, 2006. [online; last accessed 2012-03-14].
- [11] L. E. Baum and T. Petrie. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *The Annals of Mathematical Statistics*, 37(6):1554–1563, 1966.

- [12] BBC. Burglar jailed over unusual walk. http://news.bbc.co.uk/2/hi/uk_news/england/lancashire/7343702.stm. [online; last accessed 2012-03-14].
- [13] M. Bächlin, J. Schumm, D. Roggen, and G. Töster. Quantifying Gait Similarity: User Authentication and Real-World Challenge. In M. Tistarelli and M. Nixon, editors, *Advances in Biometrics*, volume 5558 of *Lecture Notes in Computer Science*, pages 1040–1049. Springer Berlin / Heidelberg, 2009.
- [14] P. N. Belhumeur, J. a. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:711–720, July 1997.
- [15] K. P. Bennett and C. Campbell. Support Vector Machines: Hype or Hallelujah? *ACM Special Interest Group on Knowledge Discovery and Data Mining Explorations*, 2, 2003.
- [16] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-García, D. Petrovska-Delacrétaz, and D. A. Reynolds. A tutorial on text-independent speaker verification. *EURASIP Journal on Applied Signal Processing*, 4: 430–451, 2004.
- [17] Biometry.com. <http://www.biometry.com>. [online; last accessed 2012-03-14].
- [18] G. E. Blonder. Graphical password, 1996.
- [19] I. Bouchrika, M. Goffredo, J. Carter, and M. Nixon. On Using Gait in Forensic Biometrics. *Journal of Forensic Sciences*, 56(4):882–889, 2011.
- [20] N. Boulgouris, D. Hatzinakos, and K. Plataniotis. Gait Recognition: A challenging signal processing technology for biometric identification. *IEEE Signal Processing Magazine*, 22(6): 78 – 90, November 2005.
- [21] N. Boulgouris, K. Plataniotis, and D. Hatzinakos. An angular transform of gait sequences for gait assisted recognition. In *IEEE International Conference on Image Processing*, pages 857–860, 2004.
- [22] P. Bours and R. Shrestha. Eigensteps: A giant leap for gait recognition. In *2nd International Workshop on Security and Communication Networks*, pages 1–6. IEEE, 2010.
- [23] H. Brandt. Classification of Acceleration Data for Biometric Gait Recognition on Mobile Devices. Master’s thesis, Hochschule Darmstadt, 2011.
- [24] F. Breitingner and C. Nickel. User Survey on Phone Security. In *BIOSIG 2010 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures*, 2010.
- [25] T. Brezmes, J.-L. Gorricho, and J. Cotrina. Activity Recognition from Accelerometer Data on a Mobile Phone. In *Distributed Computing, Artificial Intelligence, Bioinformatics, Soft Computing, and Ambient Assisted Living*, volume 5518 of *Lecture Notes in Computer Science*, pages 796–799. Springer Berlin / Heidelberg, 2009. ISBN 978-3-642-02480-1.
- [26] BTS Gaitlab. Integrated solution for multifactorial clinical gait analysis. <http://www.btsbioengineering.com/Media/Brochure/assets/BTSMED-GAITLAB-UK-WQ.pdf>. [online; last accessed 2012-03-14].

-
- [27] C. J. Burges. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Mining and Knowledge Discovery*, 2:121–167, 1998.
 - [28] M. J. Carey, E. S. Parris, and J. S. Bridle. A speaker verification system using alpha-nets. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pages 397–400, Washington, DC, USA, 1991. ISBN 0-7803-0003-3.
 - [29] P. Cattin. *Biometric Authentication System Using Human Gait*. PhD thesis, Swiss Federal Institute of Technology, 2002.
 - [30] C. H. Chan. *Multi-scale Local binary Pattern Histogram for Face Recognition*. PhD thesis, University of Surrey, 2008.
 - [31] C. H. Chan, J. Kittler, N. Poh, T. Ahonen, and M. Pietikainen. (Multiscale) Local Phase Quantisation histogram discriminant analysis with score normalisation for robust face recognition. In *IEEE 12th International Conference on Computer Vision Workshops*, pages 633–640, 27 2009-oct. 4 2009.
 - [32] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. <http://www.csie.ntu.edu.tw/~cjlin/libsvm>, 2001. [online; last accessed 2012-03-14].
 - [33] Classifeye. <http://www.classifeye.com>. [online; last accessed 2011-10-10].
 - [34] T. M. Cover. Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition. *IEEE Transactions on Electronic Computers*, 14(3): 326–334, June 1965.
 - [35] M. J. Covington, M. Ahamad, I. A. Essa, and H. Venkateswaran. Parameterized Authentication. In P. Samarati, P. Y. A. Ryan, D. Gollmann, and R. Molva, editors, *European Symposium on Research in Computer Security*, volume 3193 of *Lecture Notes in Computer Science*, pages 276–292. Springer, 2004. ISBN 3-540-22987-6.
 - [36] D. Davis, F. Monroe, and M. K. Reiter. On user choice in graphical password schemes. In *Proceedings of the 13th conference on USENIX Security Symposium*, volume 13, pages 151–164, Berkeley, CA, USA, 2004. USENIX Association.
 - [37] A. De Angeli, L. Coventry, G. Johnson, and K. Renaud. Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems. *International Journal of Human Computer Studies*, 63:128–152, July 2005.
 - [38] A. de Santos Sierra, C. Sánchez-Ávila, A. Mendaza Ormaza, and J. Guerra Casanova. An approach to hand biometrics in mobile devices. *Signal, Image and Video Processing*, 5: 469–475, 2011.
 - [39] M. O. Derawi. Multi-modal Biometric Authentication using Gait and Fingerprint on Mobile Devices. Master’s thesis, Technical University of Denmark, 2009.
 - [40] M. O. Derawi, P. Bours, and K. Holien. Improved Cycle Detection for Accelerometer Based Gait Authentication. In *6th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2010.

- [41] M. O. Derawi, C. Nickel, P. Bours, and C. Busch. Unobtrusive User-Authentication on Mobile Phones using Biometric Gait Recognition. In *6th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2010.
- [42] M. O. Derawi, B. Yang, and C. Busch. Fingerprint Recognition with Embedded Cameras on Mobile Phones. In *3rd International ICST Conference on Security and Privacy in Mobile Information and Communication Systems*, 2011.
- [43] R. Dhamija and A. Perrig. Déjà Vu: A User Study Using Images for Authentication. In *Proceedings of the 9th conference on USENIX Security Symposium*, 2000.
- [44] D. P. W. Ellis. PLP and RASTA (and MFCC, and inversion) in Matlab. <http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/>, 2005. [online; last accessed 2012-03-14].
- [45] Fordham University. Wireless Sensor Datamining Project. <http://www.cis.fordham.edu/wisdm/index.php>. [online; last accessed 2012-03-14].
- [46] J. Frank. Webpage. <http://www.cs.mcgill.ca/~jfrank8/>. [online; last accessed 2012-03-14].
- [47] J. Frank, S. Mannor, and D. Precup. Activity and Gait Recognition with Time-Delay Embeddings. In *AAAI Conference on Artificial Intelligence*, 2010.
- [48] D. Gafurov. A Survey of Biometric Gait Recognition: Approaches, Security and Challenges. In *Proceedings of Annual Norwegian Computer Science Conference*, 2007.
- [49] D. Gafurov. Security Analysis of Impostor Attempts with Respect to Gender in Gait Biometrics. In *Proceedings of IEEE International Conference on Biometrics: Theory, Applications and Systems*, pages 1–6, 2007.
- [50] D. Gafurov. *Performance and Security Analysis of Gait-based User Authentication*. PhD thesis, University of Oslo, 2008.
- [51] D. Gafurov and P. Bours. Improved Hip-Based Individual Recognition Using Wearable Motion Recording Sensor. In T.-h. Kim, W.-c. Fang, M. K. Khan, K. P. Arnett, H.-j. Kang, and D. Slezak, editors, *Security Technology, Disaster Recovery and Business Continuity*, volume 122 of *Communications in Computer and Information Science*, pages 179–186. Springer Berlin Heidelberg, 2010. ISBN 978-3-642-17610-4.
- [52] D. Gafurov, P. Bours, and E. Snekenes. User authentication based on foot motion. *Signal, Image and Video Processing*, 5:457–467, 2011.
- [53] D. Gafurov, J. Hagen, and E. Snekenes. Temporal Characteristics of Gait Biometrics. In *2nd International Conference on Computer Engineering and Applications*, volume 2, pages 557–561, 2010.
- [54] D. Gafurov, K. Helkala, and T. Søndrol. Biometric Gait Authentication Using Accelerometer Sensor. *Journal of Computers*, 1(7), 2006.

- [55] D. Gafurov, K. Helkala, and T. Søndrol. Gait Recognition Using Acceleration from MEMS. In *Proceedings of The 1st International Conference on Availability, Reliability and Security*, pages 432–439, 2006.
- [56] D. Gafurov and E. Snekkenes. Towards understanding the uniqueness of gait biometric. In *8th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 1–8, 2008.
- [57] D. Gafurov and E. Snekkenes. Gait recognition using wearable motion recording sensors. *EURASIP Journal on Advances in Signal Processing*, 2009:1–16, 2009.
- [58] D. Gafurov, E. Snekkenes, and P. Bours. Gait Authentication and Identification Using Wearable Accelerometer Sensor. In *Proceedings of the IEEE Workshop on Automatic Identification Advanced Technologies*, 2007.
- [59] D. Gafurov, E. Snekkenes, and P. Bours. Spoof Attacks on Gait Authentication System. *IEEE Transactions on Information Forensics and Security, Special Issue on "Human Detection and Recognition"*, 2, 2007.
- [60] D. Gafurov, E. Snekkenes, and P. Bours. Improved Gait Recognition Performance Using Cycle Matching. In *IEEE 24th International Conference on Advanced Information Networking and Applications Workshops*, April 2010.
- [61] D. Gafurov, E. Snekkenes, and T. E. Buvarp. Robustness of Biometric Gait Authentication Against Impersonation Attack. In *Proceedings of the 2006 international conference on On the Move to Meaningful Internet Systems*, Lecture Notes in Computer Science, pages 479–488. Springer, 2006.
- [62] D. Gafurov and E. Snekkenes. Arm Swing as a Weak Biometric for Unobtrusive User Authentication. In *Proceedings of the 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pages 1080–1087, Washington, DC, USA, 2008. IEEE Computer Society. ISBN 978-0-7695-3278-3.
- [63] M. Goffredo, I. Bouchrika, J. Carter, and M. Nixon. Self-Calibrating View-invariant Gait Biometrics. *IEEE Transactions on Systems, Man and Cybernetics B*, 40(4):997–1008, 2010.
- [64] Google. Google smartphone G1. http://www.google.com/intl/en_us/mobile/android/hpp.html.
- [65] J. Guerra-Casanova, C. S. Avila, G. Bailador, and A. de Santos-Sierra. Time series distances measures to analyze in-air signatures to authenticate users on mobile phones. In *IEEE International Carnahan Conference on Security Technology*, pages 1–7, October 2011.
- [66] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA Data Mining Software: An Update. In *ACM Special Interest Group on Knowledge Discovery and Data Mining Explorations*, volume 11, 2009.
- [67] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

- [68] C. Hinds and E. Chinedu. Increasing security and usability of computer systems with graphical passwords. In *Proceedings of the 45th annual southeast regional conference*, pages 529–530, New York, NY, USA, 2007. ISBN 978-1-59593-629-5.
- [69] K. Holien. Gait recognition under non-standard circumstances. Master’s thesis, Gjøvik University College, 2008.
- [70] K. Holien, R. Hammersland, and T. Risa. How Different Surfaces Affect Gait Based Authentication. Technical report, Norwegian Information Security Lab, Gjøvik University College, 2007.
- [71] D. T. G. Huynh. *Human Activity Recognition with Wearable Sensors*. PhD thesis, Technische Universität Darmstadt, September 2008.
- [72] International Telecommunication Union. Measuring the Information Society. Technical report, 2011.
- [73] ISO/IEC JTC1 SC37 Biometrics. *ISO/IEC SC37 SD11 General Biometric System*. International Organization for Standardization, May 2008.
- [74] W. Jansen, S. Favrla, V. Krorolev, R. Ayers, and R. Swanstrom. Picture Password: A Visual Login Technique for Mobile Devices. Technical report, National Institute of Standards and Technology, 2003.
- [75] I. Jermyn, A. Mayer, F. Monrose, M. K. Reiter, and A. D. Rubin. The design and analysis of graphical passwords. In *Proceedings of the 8th conference on USENIX Security Symposium*, volume 8, Berkeley, CA, USA, 1999.
- [76] A. Y. Johnson and A. F. Bobick. A Multi-view Method for Gait Recognition Using Static Body Parameters. In *Proceedings of the Third International Conference on Audio- and Video-Based Biometric Person Authentication*, pages 301–311, London, UK, 2001. Springer-Verlag. ISBN 3-540-42216-1.
- [77] A. A. Kale, N. P. Cuntoor, B. Yegnanarayana, A. N. Rajagopalan, and R. Chellappa. Gait Analysis for Human Identification. In *4th Conference on Audio- and Video-Based Biometric Person Authentication*, pages 706–714, 2003.
- [78] M. Kunz, K. Kasper, H. Reininger, M. Möbius, and J. Ohms. Continuous Speaker Verification in Realtime. In *BIOSIG 2011 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures*, 2011.
- [79] J. Kwapisz, G. Weiss, and S. Moore. Cell phone-based biometric identification. In *4th IEEE International Conference on Biometrics: Theory Applications and Systems*, pages 1–7, 2010.
- [80] P. K. Larsen, E. B. Simonsen, and N. Lynnerup. Gait Analysis in Forensic Medicine. *Journal of Forensic Sciences*, 53(5):1149–1153, 2008.
- [81] C. Lee, S. Lee, J. Kim, and S.-J. Kim. Preprocessing of a Fingerprint Image Captured with a Mobile Camera. In D. Zhang and A. Jain, editors, *Advances in Biometrics*, volume 3832 of *Lecture Notes in Computer Science*, pages 348–355. Springer Berlin / Heidelberg, 2005. ISBN 978-3-540-31111-9.

-
- [82] L. Lee and W. Grimson. Gait analysis for recognition and classification. In *Proceedings of the 5th IEEE International Conference on Automatic Face and Gesture Recognition*, pages 148–155, may 2002.
- [83] S. Z. Li and A. K. Jain, editors. *Encyclopedia of Biometrics*. Springer US, 2009. ISBN 978-0-387-73002-8, 978-0-387-73003-5.
- [84] B. Logan. Mel Frequency Cepstral Coefficients for Music Modeling. In *International Symposium on Music Information Retrieval*, 2000.
- [85] H. Lu, A. J. B. Brush, B. Priyantha, A. K. Karlson, and J. Liu. SpeakerSense: energy efficient unobtrusive speaker identification on mobile phones. In *Proceedings of the 9th international conference on Pervasive computing*, pages 188–205, Berlin, Heidelberg, 2011. Springer-Verlag. ISBN 978-3-642-21725-8.
- [86] A. Mahalanobis, B. V. K. VijayaKumar, S. Song, S. R. F. Sims, and J. F. Epperson. Unconstrained correlation filters. *Applied Optics*, 33(17):3751–3759, June 1994.
- [87] S. Marcel, C. McCool, P. Matějka, T. Ahonen, J. Černocký, S. Chakraborty, V. Balasubramanian, S. Panchanathan, C. H. Chan, J. Kittler, N. Poh, B. Fauve, O. Glembek, O. Plchot, Z. Jančík, A. Larcher, C. Lévy, D. Matrouf, J.-F. Bonastre, P.-H. Lee, J.-Y. Hung, S.-W. Wu, Y.-P. Hung, L. Machlica, J. Mason, S. Mau, C. Sanderson, D. Monzo, A. Albiol, H. V. Nguyen, L. Bai, Y. Wang, M. Niskanen, M. Turtinen, J. A. Nolasco-Flores, L. P. Garcia-Perera, R. Aceves-Lopez, M. Villegas, and R. Paredes. On the results of the first mobile biometry (MOBIO) face and speaker verification evaluation. In *Proceedings of the 20th International conference on Recognizing patterns in signals, speech, images, and videos*, pages 210–225, Berlin, Heidelberg, 2010. Springer-Verlag. ISBN 3-642-17710-7, 978-3-642-17710-1.
- [88] D. Matovski, M. Nixon, S. Mahmoodi, and J. Carter. The Effect of Time on Gait Recognition Performance. *IEEE Transactions on Information Forensics and Security*, 7(2):543–552, 2012.
- [89] A. Mendaza-Ormaza, O. Miguel-Hurtado, R. Blanco-Gonzalo, and F.-J. Diez-Jimeno. Analysis of handwritten signature performances using mobile devices. In *IEEE International Carnahan Conference on Security Technology*, pages 1–6, October 2011.
- [90] C. Mertens, F. Grenez, and J. Schoentgen. Preliminary evaluation of speech sample salience analysis for speech cycle detection. In *Proceedings of 3rd Advanced Voice Function Assessment International Workshop*, pages 29–32, May 2009.
- [91] L. Middleton, A. A. Buss, A. I. Bazin, and M. S. Nixon. A floor sensor system for gait recognition. In *4th IEEE Workshop on Automatic Identification Advanced Technologies*, 2005.
- [92] B. Mjaaland, P. Bours, and D. Gligoroski. Walk the Walk: Attacking Gait Biometrics by Imitation. In M. Burmester, G. Tsudik, S. Magliveras, and I. Ilic, editors, *Information Security*, volume 6531 of *Lecture Notes in Computer Science*, pages 361–380. Springer Berlin / Heidelberg, 2011.
- [93] B. B. Mjaaland and P. B. and. Gait Mimicking - Attack Resistance Testing of Gait Authentication Systems. In *The Norwegian Information Security Conference*, 2009.

- [94] Mäntyjärvi, J. and Lindholm, M. and Vildjiounaite, E. and Makela, S.-M. and Ailisto, H.A. Identifying users of portable devices from gait pattern with accelerometers. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 2, pages ii/973–ii/976, March 2005.
- [95] Mobbeel. <http://www.mobbeel.com>. [online; last accessed 2012-03-14].
- [96] R. Morris and K. Thompson. Password security: a case history. *Communications of the ACM*, 22:594–597, November 1979.
- [97] S. J. Morris. *A Shoe-Integrated Sensor System for Wireless Gait Analysis and Real-Time Therapeutic Feedback*. PhD thesis, Massachusetts Institute of Technology, June 2004.
- [98] Motorola. Motorola atrix smartphone. <http://www.motorola.com/Consumers/US-EN/Consumer-Product-and-Services/Mobile-Phones/Motorola-ATRIX-US-EN>. [online; last accessed 2011-12-12].
- [99] S. Mracek, C. Busch, R. Dvorak, and M. Drahansky. Inspired by Bertillon – Recognition Based on Anatomical Features from 3D Face Scans. In *Proceedings IEEE International Workshop on Security and Communication Networks*, 2011.
- [100] M. Müller. *Information Retrieval for Music and Motion*, chapter 4 - Dynamic Time Warping. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007. ISBN 3540740473.
- [101] D.-P. Munteanu and S.-A. Toma. Automatic speaker verification experiments using HMM. In *8th International Conference on Communications*, 2010.
- [102] M. Murray. Gait as a total pattern of movement. *American Journal of Physical Medicine & Rehabilitation*, 46(1):290–333, 1967.
- [103] M. P. Murray, A. B. Drought, and R. C. Kory. Walking Patterns of Normal Men. *The Journal of Bone and Joint Surgery*, 46:335–360, 1964.
- [104] C. K. Ng, M. Savvides, and P. K. Khosla. Real-Time Face Verification System on a Cell-Phone Using Advanced Correlation Filters. In *Proceedings of the 4th IEEE Workshop on Automatic Identification Advanced Technologies*, pages 57–62, Washington, DC, USA, 2005. ISBN 0-7695-2475-3.
- [105] C. Nickel, H. Brandt, and C. Busch. Benchmarking the Performance of SVMs and HMMs for Accelerometer-Based Biometric Gait Recognition. In *Proceedings of the 11th IEEE International Symposium on Signal Processing and Information Technology*, 2011.
- [106] C. Nickel, H. Brandt, and C. Busch. Classification of Acceleration Data for Biometric Gait Recognition on Mobile Devices. In *BIOSIG 2011 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures*, 2011.
- [107] C. Nickel and C. Busch. Classifying accelerometer data via hidden markov models to authenticate people by the way they walk. In *IEEE International Carnahan Conference on Security Technology*, pages 1–6, October 2011.

- [108] C. Nickel and C. Busch. Does a Cycle-based Segmentation Improve Accelerometer-based Biometric Gait Recognition? In *Proceedings of the 1th IEEE International Conference on Information Science, Signal Processing and their Applications*, 2012.
- [109] C. Nickel, C. Busch, S. Rangarajan, and M. Möbius. Using Hidden Markov Models for Accelerometer-Based Biometric Gait Recognition. In *7th International Colloquium on Signal Processing & Its Applications*, 2011.
- [110] C. Nickel, M. O. Derawi, P. Bours, and C. Busch. Scenario Test of Accelerometer-Based Biometric Gait Recognition. In *3rd International Workshop on Security and Communication Networks*, 2011.
- [111] C. Nickel, T. Wirtl, and C. Busch. Authentication of Smartphone Users Based on the Way They Walk Using k-NN Algorithm. In *Proceedings of the 8th IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2012.
- [112] C. Nickel, X. Zhou, and C. Busch. Template Protection for Biometric Gait Data. In *BIOSIG 2010 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures*, 2010.
- [113] R. S. Nickerson. Short-term memory for complex meaningful visual configurations: A demonstration of capacity. *Canadian Journal of Psychology/Revue canadienne de psychologie*, 19(2):155–160, 1965.
- [114] M. S. Nixon, J. N. Carter, J. M. Nash, P. S. Huang, D. Cunado, and S. V. Stevenage. Automatic gait recognition. In *Biometrics - Personal Identification in Networked Society*, pages 231–250. Kluwer, 1999.
- [115] OMRON Corporation. http://www.omron.com/media/press/2005/02/n_280205.html. [online; last accessed 2012-03-14].
- [116] A. V. Oppenheim and R. W. Schaffer. *Discrete-Time Signal Processing*. Prentice Hall Press, Upper Saddle River, NJ, USA, 3rd edition, 2009. ISBN 0131988425, 9780131988422.
- [117] ORACLE. Opportunity Calling: The Future of Mobile Communications – Take Two, 2011.
- [118] R. J. Orr and G. D. Abowd. The smart floor: a mechanism for natural user identification and tracking. In *Extended Abstracts of the Conference on Human Factors in Computing Systems*, pages 275–276, New York, NY, USA, 2000. ISBN 1-58113-248-4.
- [119] A. J. O’Toole, P. J. Phillips, S. Weimer, D. A. Roark, J. Ayyad, R. Barwick, and J. Dunlop. Recognizing people from dynamic and static faces and bodies: Dissecting identity with a fusion approach. *Vision Research*, 51(1):74–83, 2011.
- [120] G. Pan, Y. Zhang, and Z. Wu. Accelerometer-based gait recognition via voting by signature points. *Electronics Letters*, 45(22):1116–1118, October 2009.
- [121] Pantech. GI100 smartphone. <http://www.dexigner.com/news/5242>. [online; last accessed 2012-03-14].
- [122] L. Rabiner and B. Juang. An introduction to hidden markov models. *IEEE acoustics, speech, and signal processing magazine*, 3(1):4–16, April 1986.

- [123] P. Radu, K. Sirlantzis, W. Howells, F. Deravi, and S. Hoque. Information Fusion for Unconstrained Iris Recognition. In *International Journal of Hybrid Information Technology*, volume 4, October 2011.
- [124] D. Ritter. Bildbasierte zugriffskontrolle für passwortverwaltungssysteme auf mobilen geräten, 2011.
- [125] L. Rong, Z. Jianzhong, L. Ming, and H. Xiangfeng. A Wearable Acceleration Sensor System for Gait Recognition. In *2nd IEEE Conference on Industrial Electronics and Applications*, pages 2654–2659, May 2007.
- [126] L. Rong, D. Zhiguo, Z. Jianzhong, and L. Ming. Identification of Individual Walking Patterns Using Gait Acceleration. In *The 1st International Conference on Bioinformatics and Biomedical Engineering*, pages 543–546, July 2007.
- [127] A. Roy, M. Magimai-Doss, and S. Marcel. Fast Speaker Verification on Mobile Phone data using Boosted Slice Classifiers. In *IEEE International Joint Conference on Biometrics*, 2011.
- [128] A. Schlapbach and H. Bunke. Using HMM Based Recognizers for Writer Identification and Verification. In *Proceedings of the 9th International Workshop on Frontiers in Handwriting Recognition*, pages 167–172, 2004. ISBN 0-7695-2187-8.
- [129] SFR Sorftware GmbH. Visual Key Technology. <http://www.sfr-software.de/cms/DE/enterprise/technik/>. [online; last accessed 2012-03-14].
- [130] Y. Shoji, A. Itai, and H. Yasukawa. Personal Identification Using Footstep Detection in In-Door Environment. *Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, E88-A:2072–2077, August 2005.
- [131] T. Søndrol. Using the Human Gait for Authentication. Master’s thesis, Gjøvik University College (NISlab), June 2005.
- [132] S. Sprager. A Cumulant-Based Method for Gait Identification Using Accelerometer Data with Principal Component Analysis and Support Vector Machine. In *Proceedings of the 2nd WSEAS International Conference on Sensors, Signals, Visualization, Imaging, Simulation and Materials*, pages 94–99, 2009.
- [133] O. Stang. Gait analysis: Is it possible to learn to walk like someone else? Master’s thesis, Gjøvik University College (NISlab), 2007.
- [134] C. Stein and C. Nickel. Eignung von Smartphone-Kameras zur Fingerabdruckerkennung und Methoden zur Verbesserung der Qualität der Fingerbilder. In *BIOSIG 2011 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures*, 2011.
- [135] S. S. Stevens, Je, and E. B. Newman. A scale for the measurement of the psychological magnitude of pitch. *Journal of the Acoustical Society of America*, 8:185–190, 1937.
- [136] STMicroelectronics, http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/DATASHEET/CD00213470.pdf. *MEMS digital output motion sensor ultra low-power high performance 3-axes nano accelerometer (LIS 331 DLH)*. [online; last accessed 2012-03-14].

-
- [137] Q. Su, J. Tian, X. Chen, and X. Yang. A fingerprint authentication mobile phone based on sweep sensor. In S. Singh, M. Singh, C. Apte, and P. Perner, editors, *Pattern Recognition and Image Analysis*, volume 3687 of *Lecture Notes in Computer Science*, pages 295–301. Springer Berlin / Heidelberg, 2005. ISBN 978-3-540-28833-6.
- [138] J. Suutala, K. Fujinami, and J. Rönning. Gaussian Process Person Identifier Based on Simple Floor Sensors. In *Proceedings of the 3rd European Conference on Smart Sensing and Context*, pages 55–68, Berlin / Heidelberg, 2008. Springer-Verlag. ISBN 978-3-540-88792-8.
- [139] J. Suutala and J. Rönning. Methods for person identification on a pressure-sensitive floor: Experiments with multiple classifiers and reject option. *Journal for Information Fusion*, 9(1):21–40, January 2008.
- [140] T. Takada and H. Koike. Awase-E: Image-Based Authentication for Mobile Phones Using User’s Favorite Images. In L. Chittaro, editor, *Human-Computer Interaction with Mobile Devices and Services*, volume 2795 of *Lecture Notes in Computer Science*, pages 347–351. Springer Berlin / Heidelberg, 2003. ISBN 978-3-540-40821-5.
- [141] J. Thorpe and P. C. v. Oorschot. Towards Secure Design Choices for Implementing Graphical Passwords. In *Proceedings of the 20th Annual Computer Security Applications Conference*, pages 50–60, Washington, DC, USA, 2004. ISBN 0-7695-2252-1.
- [142] G. Trivino, A. Alvarez-Alvarez, and G. Bailador. Application of the computational theory of perceptions to human gait pattern recognition. *Pattern Recognition*, 43(7):2572–2581, July 2010.
- [143] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3: 71–86, January 1991.
- [144] S. Umar and Q. Rafiq. A Graphical Interface for User Authentication on Mobile Phones. In *The 4th International Conference on Advances in Computer-Human Interactions*, 2011.
- [145] University of South Florida. Human ID Gait Challenge Problem. <http://marathon.csee.usf.edu/GaitBaseline/>. [online; last accessed 2012-03-14].
- [146] University of Southampton. Automatic Gait Recognition for Human ID at a Distance. <http://www.gait.ecs.soton.ac.uk/>. [online; last accessed 2012-03-14].
- [147] E. Vildjiounaite, S.-M. Makela, M. Lindholm, V. Kyllonen, and H. Ailisto. Increasing Security of Mobile Devices by Decreasing User Effort in Verification. In *2nd International Conference on Systems and Networks Communications*, August 2007.
- [148] E. Vildjiounaite, S.-M. Mäkelä, M. Lindholm, R. Riihimäki, V. Kyllönen, J. Mäntyjärvi, and H. Ailisto. *Unobtrusive Multimodal Biometrics for Ensuring Privacy and Information Security with Personal Devices*, pages 187–201. 2006.
- [149] M. Villegas and R. Paredes. On Optimising Local Feature Face Recognition for Mobile Devices. In *V Jornadas de Reconocimiento Biométrico de Personas*, 2010.
- [150] P. Viola and M. J. Jones. Robust Real-Time Face Detection. *International Journal of Computer Vision*, 57:137–154, May 2004.

- [151] A. Viterbi. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, 13(2):260–269, April 1967.
- [152] X. Wang, Y. Li, and F. Qiao. Gait authentication based on multi-criterion model of acceleration features. In *International Conference on Modelling, Identification and Control*, pages 664–669, 2010.
- [153] Y. Wang, F. Zhang, and L. Chen. An Approach to Incremental SVM Learning Algorithm. In *ISECS International Colloquium on Computing, Communication, Control, and Management*, 2008.
- [154] S. Wiedenbeck, J. Waters, J.-C. Birget, A. Brodskiy, and N. Memon. Authentication using graphical passwords: effects of tolerance and image choice. In *Proceedings of the 2005 symposium on usable privacy and security*, pages 1–12, New York, NY, USA, 2005. ISBN 1-59593-178-3.
- [155] S. Wiedenbeck, J. Waters, J.-C. Birget, A. Brodskiy, and N. Memon. Passpoints: design and longitudinal evaluation of a graphical password system. *International Journal of Human Computer Studies*, 63:102–127, July 2005.
- [156] D. Winter. *Biomechanics and Motor Control of Human Movement*. Wiley, 2009. ISBN 9780470398180.
- [157] H. Witte and C. Nickel. Modular Biometric Authentication Service System (MBASSy). In *BIOSIG 2010 - Proceedings of the Special Interest Group on Biometrics and Electronic Signatures*, 2010.
- [158] K. Xi, Y. Tang, J. Hu, and F. Han. A correlation based face verification scheme designed for mobile device access control: From algorithm to Java ME implementation. In *5th IEEE Conference on Industrial Electronics and Applications*, pages 317–322, June 2010.
- [159] K. Yamauchi, B. Bhanu, and H. Saito. Recognition of walking humans in 3D: Initial results. *Computer Vision and Pattern Recognition Workshop*, pages 45–52, 2009.
- [160] L. Yan, L. Yue-e, and H. Jian. Gait Recognition Based on MEMS Accelerometer. In *10th International Conference on Signal Processing*, pages 1679–1681, 2010.
- [161] S. J. Young, G. Evermann, M. J. F. Gales, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Valtchev, and P. C. Woodland. *The HTK Book, version 3.4*. Cambridge University Engineering Department, Cambridge, UK, 2006.
- [162] J. Yun. User Identification Using Gait Patterns on UbiFloorII. *Journal of Sensors*, 11(3): 2611–2639, 2011.
- [163] J.-S. Yun, Seung-HunLee, W.-T. Woo, and J.-H. Ryu. The User Identification System USING Walking Pattern over the ubiFloor. In *Proceedings of the ICCAS*, 2003.
- [164] E. Zwicker. Subdivision of the Audible Frequency Range into Critical Bands (Frequenzgruppen). *The Journal of the Acoustical Society of America*, 33(2), 1961.